NIKHEF 96-017

Multidimensional sampling for simulation and integration:

measures, discrepancies, and quasi-random numbers

F. James[1]
CERN, Geneva, Switzerland
J. Hoogland[2]
NIKHEF-H, Amsterdam, the Netherlands
R. Kleiss[3]
University of Nijmegen, Nijmegen, the Netherlands

Abstract

This is basically a review of the field of Quasi-Monte Carlo intended for computational physicists and other potential users of quasi-random numbers. As such, much of the material is not new, but is presented here in a style hopefully more accessible to physicists than the specialized mathematical literature. There are also some new results: On the practical side we give important empirical properties of large quasi-random point sets, especially the exact quadratic discrepancies; on the theoretical side, there is the exact distribution of quadratic discrepancy for random point sets.

Preprint accepted for publication in Computer Physics Communications

[1]e-mail: F.James@cern.ch
[2]e-mail: t96@nikhef.nl, research supported by the Stichting FOM
[3]e-mail: kleiss@sci.kun.nl

1 Introduction

In numerical integration, the magnitude of the integration error depends on two things:

1. The fluctuating behaviour(variance,variation) of the integrand. This behaviour may be improved by so-called *variance-reduction techniques*, which we shall not discuss in this paper (see, for example,[3]).

2. The quality of the point-set employed for the integration. We shall see that the measure of quality will be the uniformity of the point-set. This paper deals with the uniformity properties of point sets for numerical integration: in particular, we discuss various definitions of uniformity, their relation to typical integrands encountered, some of their theoretical properties, and the actual performance of algorithms (quasi-random number generators) claimed to yield very uniform point sets.

The outline of this paper is as follows: in section 2, we discuss the traditional Monte Carlo approach to numerical integration and relate the relatively slow convergence of the error, as $1/\sqrt{N}$ (where N denotes the size of the point set), to the inherent non-uniformity of the truly random points, used in classical Monte Carlo. This is made explicit by the notion of *classical discrepancy* of the point set; we quote several theorems relating this discrepancy to the integration error. From these it is seen, that low-discrepancy point sets lead to small errors. In section 3, we describe a number of algorithms that have been proposed for constructing *quasi-random sequences*, that lead to point sets with a discrepancy smaller than that expected for truly random point sets. In general, for these algorithms, there exist discrepancy bounds, that are valid only in the limit $N \to \infty$. (Their observed behaviour for finite N is reported in section 6.) In section 4, we point out that the definition of classical discrepancy is intimately connected with the underlying, implicit, model for the type of integrand attacked; this *function class* is governed by the Wiener measure, which may or may not be appropriate, depending on the context of the integration problem. We describe a number of alternative choices of function class, each leading to its own definition of discrepancy. In section 5, we argue that the merit of a 'low-discrepancy' sequence can only be judged in relation to a truly random point set. To this end, one has to know the probability distribution of a given discrepancy, viewed as a stochastic variable through its dependence of random points. We derive several results in this direction. Finally, in section 6, we compute the classical discrepancy for a number of quasi-random sequences for moderately large point sets (N up to 150000), and moderately large dimension (up to 20).

2 Integration and discrepancy

2.1 Numerical integration

The problem we shall be considering is the estimation of the multidimensional integral

$$J = \int_K d\mu(x)\ f(x) \quad , \tag{1}$$

where $x = x^\mu = (x^1, x^2, \ldots, x^s)$ denotes a point in the s-dimensional integration region K. Throughout this paper, Greek indices like μ will denote individual coordinates. The integrand is $f(x)$, and $d\mu(x)$ denotes a measure on K, that is, we must have

$$\int_K d\mu(x) = 1 \quad , \tag{2}$$

and $d\mu(x)$ must be positive. We shall only consider measures that allow a probabilistic interpretation, that is, there exists, *in principle,* a computer algorithm for turning a sequence of truly random numbers into a series of points x such that their probability density is given by $d\mu(x)$:

$$\mathrm{Prob}(x \in A) = \int_A d\mu(x) \quad , \tag{3}$$

for all small rectangular regions A.

We estimate J using the unweighted sum:

$$S = \frac{1}{N} \sum_k f(x_k) \quad , \tag{4}$$

where the x_k constitute a *point set,* that is, a finite set of N points x_k, $k = 1, 2, \ldots, N$ obtained in one or another way. Note that Latin indices like k run over the points in the point set and the sum is always understood to run from 1 to N. The error made in using this estimate is clearly

$$\eta = S - J \quad . \tag{5}$$

The object of this paper is to find point sets x_k which can be used to estimate the integrals of general classes of functions f with small error η.

We restrict the choice of integration regions K to the unit hypercube $I_s = [0, 1)^s$ since this is the most common and best understood region, but we should point out that the properties of many integration methods may depend strongly on the shape of the integration region.

Finally, throughout this paper we shall adopt the real-number model of computation, that is, we shall disregard all kinds of small corrections related to the finite word size of our computers, or the fact that the numbers we use cannot be truly irrational: we assume that our computers have enough precision to approximate real numbers for all practical purposes.

2.2 Monte Carlo integration: the Central Limit Theorem

In the Monte Carlo method the point set is taken to be a sample of random points, independent and identically distributed (*iid*) with probability density $d\mu(x)$. The error η is then also a random quantity, with some probability density $P(\eta)$. This is most easily studied in the form of a *moment-generating function* G: with $M = \sqrt{N}$, we can write

$$G(Mz) \equiv \left\langle\, e^{Mz\eta} \,\right\rangle = \left\langle\, e^{-MzJ} \prod_k e^{zf(x_k)/M} \,\right\rangle \quad , \tag{6}$$

where the brackets denote an average over all point sets consisting of N points chosen with the measure $d\mu(x)$ as a probability density. Since the points x_k are random, they are independent, and we can easily take the average:

$$\left\langle\, e^{zf(x_k)/M} \,\right\rangle = \sum_{n\geq 0} \frac{z^n}{n!M^n} J_n \quad , \quad J_n \equiv \int\limits_K d\mu(x)\; f(x)^n \quad . \tag{7}$$

For large enough N, the following approximation is justified:

$$\begin{aligned} G(Mz) &= \left[e^{-zJ/M} \sum_{n\geq 0} \frac{z^n}{n!M^n} J_n \right]^N \\ &= \left[1 + \frac{z^2}{2N}(J_2 - J^2) + \frac{z^3}{6NM}(J_3 - 3J_2J + 2J^3) + \cdots \right]^N \\ &= \exp\left(\frac{z^2}{2}(J_2 - J^2) \right) (1 + \mathcal{O}(1/M)) \quad . \end{aligned} \tag{8}$$

Up to terms of order $1/M$, we may compute $P(\eta)$ by Fourier transform:

$$\begin{aligned} P(\eta) &= \frac{1}{2\pi} \int\limits_{-\infty}^{\infty} dz\; e^{-iz\eta} G(iz) \;\sim\; \frac{1}{\sqrt{2\pi V/N}} \exp\left(\frac{-N\eta^2}{2V} \right) \quad , \\ V &= J_2 - J^2 \quad . \end{aligned} \tag{9}$$

This is the Central Limit Theorem: the error is normally distributed around zero, with standard deviation $\sqrt{V/N}$; V is called the *variance* of the integrand: it is positive for every $f(x)$ that is not constant over K. We note the following aspects of the above derivation.

- The Monte Carlo estimate is *unbiased*: the expectation value of the error is zero (this holds also for finite N).
- The error distribution attains a simple form owing to the 'large-N approximation', that is, we neglect terms of order $1/N$.

- The result for the expected square error is rigorous: but its use as an estimate for the error made using *this particular point set* implies that we postulate this point set to be a 'typical' member of the ensemble of point sets governed by the underlying measure $d\mu(x)$.
- The error estimate does not depend on the particularities of the point set actually used, since we have integrated over all such point sets.
- The error estimate *does* depend on the particularities of the integrand, namely its variance, which is a measure of the amount by which it fluctuates over K.
- Any quadratically integrable function can be integrated by Monte Carlo but the convergence of the error to zero is (only) as $1/\sqrt{N}$, and additional smoothness properties, such as continuity or differentiability only lead to smaller error inasmuch as they lead to a smaller V.

2.3 Classical discrepancy

In the previous section we have established the $1/\sqrt{N}$ convergence of classical Monte Carlo, but it is known that, in one dimension, even the unweighted quadrature method known as the trapezoid rule, where the points are equidistantly distributed, gives a $1/N^2$ convergence. Clearly, the irregularities in the random, rather than equidistant, point set play a rôle in the behaviour of the error. We now introduce the *classical discrepancy*, which will allow us to quantify such irregularities, not only in one dimension, but also in higher dimensions where the trapezoid rule is outperformed by Monte Carlo.

Let K be the s-dimensional hypercube I_s, and y be a point in K. We also assume that $d\mu(x)$ is the usual Cartesian measure on K: random points under this $d\mu(x)$ have a uniform distribution over K. We define the following counting function:

$$\chi(y;x) = \prod_{\mu} \theta(y^\mu - x^\mu) \quad , \tag{10}$$

which simply checks if the point x is 'below y', *i.e.* inside the hyper-rectangle defined by the origin and y. The *local discrepancy* at y, for the point set x_k, is then

$$g(y) = \frac{1}{N}\sum_k \chi(y;x_k) - \prod_{\mu} y^\mu \quad . \tag{11}$$

The function $g(y)$ counts the fraction of the point set is below y, and compares this with the volume of K that is below y. From a slightly different point of view, $g(y)$ is nothing but the integration error made in estimating the volume below y using the given point set. Clearly, the more uniformly distributed the point set, the smaller $g(y)$. This suggests the notion *global discrepancy*, that must inform us about the global deviation of $g(y)$ from zero. There exist various

such quantities. Let us define

$$D_m \equiv \int_K dy\ g(y)^m \quad . \tag{12}$$

Then, useful measures of global discrepancy are D_1 (linear discrepancy), D_2 (quadratic discrepancy), and the extreme, or Kolmogorov discrepancy:

$$D_\infty = \lim_{k\to\infty} (D_{2k})^{1/2k} = \sup_{y\in K} |g(y)| \quad . \tag{13}$$

This last discrepancy (called $D^\star$ in [6]), has been the most widely studied, but D_2 appears to be easier to evaluate for a given point set (as we shall see), and may be more relevant as well.

We shall derive some results on the discrepancies for random points later on, but at this point we may already quote the various expectations for truly random point sets:

$$\langle\ D_1\ \rangle = 0\ \ ,\ \ \langle\ D_2\ \rangle = \frac{1}{N}\left(\frac{1}{2^s} - \frac{1}{3^s}\right)\ \ ,\ \ \langle\ D_\infty\ \rangle \stackrel{s=1}{=} \sqrt{\frac{\pi}{2N}}\log 2 \quad . \tag{14}$$

2.4 Results for integration errors

It should be intuitively clear that the size of the discrepancy of a point set must relate to how good that point set is for numerical integration. This has indeed been studied extensively; see, for example [6] from which we give some important results.

In the first place, if a point set has extreme discrepancy equal to D, every rectangular region (with edges parallel to the axes) of volume $2^s D$ or larger must contain at least one point[1]: the largest rectangular 'gap' (with no points) has size $2^s D_\infty$. The discrepancy is seen to measure the 'resolution' of the point set. Secondly, let us define the 'extremum metric' d on K by

$$d(x,y) = \sup_{\mu=1,2,\ldots,s} |x^\mu - y^\mu| \quad , \tag{15}$$

for any two points x and y in K. The *modulus of continuity* M of the integrand is then the following function:

$$M(z) = \sup_{x,y\in K} |f(x) - f(y)| \quad , \quad \text{for } d(x,y) \le z \quad . \tag{16}$$

That is, the modulus of continuity tells us what is the maximum jump in the value of f, if we make a step of maximal size z. Then, when using the point set to estimate J by the quantity S, the error η obeys

$$|\eta| \le 4M(D_\infty^{1/s}) \quad ; \tag{17}$$

[1]The factor 2^s is due to the restriction, in the definition of discrepancy, to hyper-rectangles touching the origin.

the appealing result that the error depends on the maximal jump in value, for steps of half the size of the edge of the hyper-cubic 'gap' that corresponds to the resolution of the point set.

Although this result is only valid for continuous functions, for which $M(z) \to 0$ as $z \to 0$, we may always approximate a discontinuous function by a continuous one: but then, $M(z)$ will no longer vanish with z, and we have a finite error bound even for zero discrepancy.

A large number of other, similar results are given in [6], for instance the Koksma-Hlawka bound which relates the error to the *variation* of the integrand, rather than to its variance: but, since for discontinuous functions the variation is in many cases infinite even when the variance is finite, such a bound may be of limited value; moreover, quantities like the modulus of continuity or the variation are in practice very hard to compute exactly. Nonetheless, these results teach us that the lower the discrepancy, the smaller we can expect the integration error to be.

2.5 Quasi-random Sampling for Simulation

Formally at least, most simulations are in fact equivalent to numerical integrations, since one is usually looking for the expectations of particular variables or of distributions of variables and these expectations are simply integrals over some subspaces (histogram bins) of the full sampling space. However, in practice, additional considerations may be important. In simulations of particle physics experiments for example, the dimensionality as defined by random numbers per event tends to be extremely high and often is not even the same from event to event. Clearly the results of this paper are not then immediately applicable, but one somehow suspects that there must nonetheless be a way to take advantage of the improved uniformity of quasi-random points in such simulations.

A promising way to approach this problem is to decompose the full simulation of an event into manageable parts, some of which may then be of reasonably low fixed dimension. For example, the physical event must be generated, and independently some of the particles may interact or decay, and independently they may produce detector hits, showers, etc. It may be advantageous to use quasi-random points in the simulation of one or more of these sub-processes.

Another difficulty in simulation is that the detector boundaries or decision boundaries tend to be discontinuous, which formally makes the problem equivalent to the numerical integration of a multidimensional discontinuous function, and the variation of such a function is generally unbounded. Here again, practical experience indicates that the situation is not as grim as all that, and that such integrals do in fact converge with a considerably smaller error using quasi-random points [9].

3 Low-discrepancy point sets

3.1 Finite sets and infinite sequences

Much effort has been devoted to construct point sets that have lower discrepancy than those consisting of random points. We must distinguish between two cases. On the one hand, one may consider *finite* point sets, where N is fixed. In principle, for each value of N in the s-dimensional hypercube, there exists a point set with the lowest discrepancy of all point sets. That discrepancy must be larger than the so-called Roth bound [6]:

$$D_\infty \geq C_s \frac{(\log N)^{(s-1)/2}}{N} \quad , \tag{18}$$

where C_s depends only on s. For large N and reasonable s, this lower bound is much lower than the expected discrepancy for random point sets but, except in one dimension, it is not known how to construct such an 'optimal' set, and it is not even known if it is possible to attain this bound in general. The one-dimensional exception is the equidistant point set:

$$x_k = \frac{2k-1}{2N} \quad , \quad k = 1, 2, \ldots, N \quad , \tag{19}$$

or any rearrangement of this set. In higher dimension, Cartesian products of this sequence, the so-called *hyper-cubic lattice,* are *not* optimal. For, let us enumerate a hyper-cubic sequence by

$$x^\mu_{(k^1,k^2,\ldots,k^s)} = \frac{2k^\mu - 1}{2M} \quad , \quad k^\mu = 1, 2, \ldots, M \quad , \quad M^s = N \quad . \tag{20}$$

For this point set, we indeed have $D_1 = 0$, but

$$D_2 = \frac{1}{3^s}\left[1 + \left(1 + \frac{1}{2M^2}\right)^s - 2\left(1 + \frac{1}{8M^2}\right)^s\right] \quad , \tag{21}$$

which only goes as $N^{-2/s}$ for large N. This discrepancy is larger than that expected for random points if $s > 2$: and this is the underlying reason why, in higher dimension, trapezoid rules generally perform less well than Monte Carlo for the same number of points.

It is in principle possible to use numerical minimization techniques to find approximately optimal point sets, but the computational complexity of this problem is so enormous that it can be attempted only for very small values of both N and s.

On the other hand, we usually do not wish to restrict ourselves in advance to a particular value of N, but rather prefer to find a formula which yields a long sequence of points for which the discrepancy is low for all N, perhaps even approaching an optimal value asymptotically. This allows us to increase the size of our point set at will until the desired convergence is attained. We shall, therefore, mainly concentrate on sequences of indefinite length. Their discrepancy for any given N will usually be larger than that of an optimal point set with the same N; but, 'on the average', we may hope to remain close to the Roth bound.

3.2 Richtmyer sequences

Probably the first quasi-random point generator was that attributed to Richtmyer and defined by the formula for the μth coordinate of the kth point:

$$x_k^\mu = [kS_\mu] \mathrm{mod}\ 1 \quad , \tag{22}$$

where the S_μ are constants which should be irrational numbers in order for the period to be infinite. Since neither irrational numbers nor infinite periods exist on real computers, the original suggestion was to take S_μ equal to the square root of the μth prime number, and this is the usual choice for these constants.

Figure 1 shows how the distribution of Richtmyer points in two dimensions evolves as N increases. A very regular band structure develops. The bands then become wider until they completely fill the empty space. Then they start to overlap and begin to form other bands in other directions which also widen, and the process continues with each successive band structure becoming narrower and more complex. This kind of behaviour is a general property of quasi-random sequences, even those described below which are based on a very different algorithm.

This simple quasi-random generator does not seem to have received much attention, even though it is the only one we know based on a linear congruential algorithm (others are based on radical-inverse transformations), and as we shall see below, its discrepancy is surprisingly good. From a practical point of view, some potential users have probably been discouraged by plots of points in two-dimensional projections which show some long-lasting band structure in some dimensions, due of course to an unfortunate combination of constants S_μ. In this paper we use always the square roots of the first prime numbers, but we should point out that it may be possible to find other values of S_μ which assure a long period and minimize the band structure or even minimize directly the discrepancy. This is of course a very compute-intensive calculation.

3.3 Van der Corput (or Halton) sequences

An interesting low-discrepancy sequence can be found as follows [1]. Let us start, in one dimension, by choosing a base, an integer b. Any integer n can then be written in base b:

$$n = n_0 + n_1 b + n_2 b^2 + n_3 b^3 + \cdots \quad . \tag{23}$$

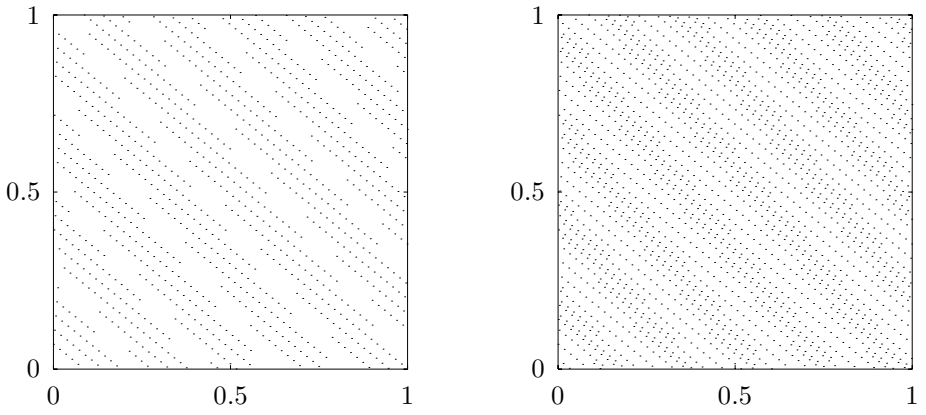
The *radical-inverse transform* (to base b) is defined by

$$\phi_b(n) = n_0 b^{-1} + n_1 b^{-2} + n_2 b^{-3} + n_3 b^{-4} + \cdots \quad . \tag{24}$$

The *van der Corput sequence* to base b is then simply the following:

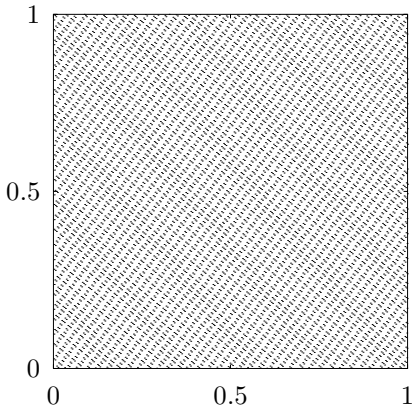
$$x_k = \phi_b(k) \quad , \quad k = 1, 2, \ldots \quad . \tag{25}$$

Note that, as k increases, the digit n_0 changes most rapidly, the digit n_1 the next rapidly, and so on: so that the leading digit in x_k changes the most rapidly,

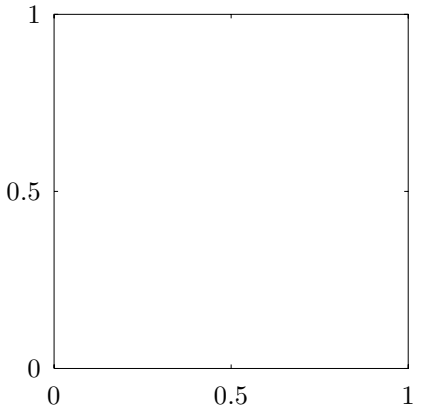


(a) 1000 points

(a) 2000 points



(a) 10000 points



(a) 40000 points

Figure 1: The distribution of Richtmyer points in two dimensions, showing the evolution as N increases.

the next-to-leading one less rapidly, and so on. It is therefore clear that this sequence manages to fill the unit interval $(0,1)$ quite efficiently. Another insight may be obtained from the fact that, when $N = b^m$, for integer m, the set $x_1, x_2, \ldots, x_N$ will precisely be equidistant, of the type of Eq.(19), and for that N the discrepancy will be optimal. The discrepancy is therefore optimal whenever N is a power of b. For other values of N, it cannot stray away too much from the optimal value, but as m increases it can go further and further out before returning: it is therefore no surprise that the discrepancy has a low upper bound:

$$D_\infty \leq C_b \frac{\log N}{N} \quad , \quad C_b = \begin{cases} \frac{b^2}{4(b+1)\log b} & \text{when } b \text{ is even} \\ \\ \frac{b-1}{4\log b} & \text{when } b \text{ is odd} \end{cases} \tag{26}$$

A generalization of Eq.(25) to higher dimensions is obvious: one simply chooses several bases $b_1, b_2, \ldots, b_s$, and

$$x_k^\mu = \phi_{b_\mu}(k) \quad , \quad k = 1, 2, \ldots \quad , \quad \mu = 1, 2, \cdots, s \quad . \tag{27}$$

In figure 2, we show the distribution, in two dimensions, of the first 1000 points obtained with $b_1 = 2$, $b_2 = 3$. This distribution is, even visually, smoother and more uniform than a typical set of 1000 random points (see figure 8). Clearly we will get into trouble if any two bases have common factors, and indeed the most common choice (and our choice here) is to take as bases the first s prime numbers. However, for dimensions 7 and 8 for instance, we then need bases 17 and 19, for which the plot is given in the bottom half of figure 2: a lot of structure is still visible, and the square is not filled as uniformly.

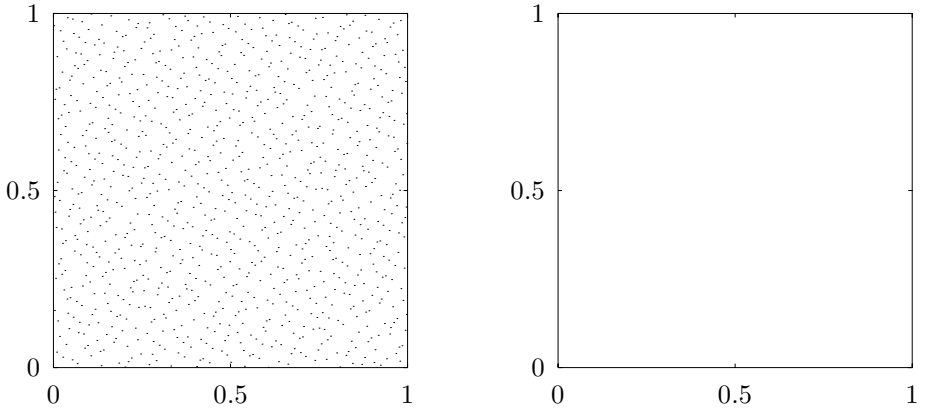
This can also be understood easily: the 'recurrence' of the discrepancy to a low value, discussed for $s = 1$, could be encountered here only if N is an exact power of each of the bases, and since these are all relatively prime, this cannot happen. Numbers that are 'close' to exact powers of all the bases (in the sense mentioned above) are guaranteed to be far apart, and hence the discrepancy can become considerable. This is in fact the content of the multidimensional equivalent of Eq.(26):

$$D_\infty \leq \frac{s}{N} + \frac{(\log N)^s}{N} \prod_\mu \left(\frac{b_\mu - 1}{2\log b_\mu} + \frac{b_\mu + 1}{2\log N} \right) \quad . \tag{28}$$

As a function of s, therefore, the constant factor grows faster than an exponential of s, and in high dimensions the superiority of the van der Corput sequence over a purely random one will appear only for impractically large N.

3.4 Sobol' and Niederreiter sequences

Improved discrepancy in higher dimensions can be obtained from the van der Corput sequences by making use of additional functions $m_\mu(k)$, $\mu = 1, 2, \ldots, s$,



(a) 1000 points

(b) 40000 points

(c) 1000 points

(d) 40000 points

Figure 2: The first van der Corput points with bases 2 and 3 (top) and bases 17 and 19 (bottom).

and defining a new sequence

$$x_k^\mu = \phi_b(m_\mu(k)) \quad , \tag{29}$$

The functions $m_\mu(k)$ are cleverly chosen, amongst other things in such a way that when $N = b^m$, the numbers $m_\mu(1), m_\mu(2), \ldots, m_\mu(N)$ are just a permutation of $(1, 2, \ldots, N)$. In this way, one can attain a constant factor (C_s in Eq.(18)) that in principle *decreases* super-exponentially with s. Practical experience with these sequences is still limited, however.

We have investigated empirically the behaviour of two such sequences: the Sobol' sequence, described in [10], and the Niederreiter base-two sequence, described in [7]. The general Niederreiter sequences form a large class of which both the above are special cases.

4 Problem classes and discrepancies

4.1 A conceptual problem for special point sets

We have seen how, in classical Monte Carlo, the error estimate is based on the assumption that the point set is one out of an ensemble of such, essentially equivalent, point sets: we then average over this ensemble. We have also seen that the lower a point set's discrepancy, the smaller the error could be. But when we try to apply this statement to the integral of a particular function, we immediately run into trouble. Namely, low-discrepancy point sets are not 'typical', they are special. In particular, the points are correlated in a possibly very complicated way, so that the simple treatment of Eq.(7) cannot be valid. One way out would be to define, in some manner, an ensemble of point sets with a given discrepancy. This approach is followed in [5].

Another option, which we shall pursue in the following, is to respect the specialness of the point set, and instead consider the integrand $f(x)$ to be a 'typical' member in some set: this we shall call the *problem class*. An example of such a class could be that of all polynomials of some degree, but we shall encounter other, more relevant, cases. Then, we may hope, instead of averaging over the ensemble of point sets to end up with an error estimate depending on the integrand, to integrate (or average) over the problem class: in that case, we will end up with an error estimate that is independent of the particular integrand, but related to some property of the point set, for instance its discrepancy as discussed above. It should be clear that different problem classes will lead to different measures of irregularity of the point set: we shall call such a measure the *induced discrepancy* corresponding to the problem class.

To pursue this line of reasoning, we must therefore first consider what constitutes a problem class, and this we shall do next.

4.2 Green's functions and connected Green's functions

It is our aim to determine the integration error by averaging over all integrands in a given problem class. Strictly speaking, we then have to define a measure

$\mathcal{D}f$ on this problem class, which defines the probability density for members of the class. In fact, we can settle for less: the kind of information we need about the problem class is the set of all *Green's functions* in the problem class:

$$\begin{aligned} g_n(x_1, x_2, \ldots, x_n) &\equiv \langle\, f(x_1)f(x_2)\cdots f(x_n)\,\rangle_f \\ &\equiv \int \mathcal{D}f\; f(x_1)f(x_2)\cdots f(x_n) \quad , \end{aligned} \tag{30}$$

where this time the brackets denote an average over the problem class: it is the 'typical' value expected for the product of integrand values at the points $x_1, x_2, \ldots, x_n$. It is, of course, symmetric in all its arguments. Knowledge of the Green's functions will be sufficient for our purposes, even if we cannot (as in the case of the Wiener measure) write down the form of a particular integrand in the problem class in any simple manner. The kind of problem class we consider, and therefore the Green's functions, must of course be determined by the nature of our integration problem. The restriction to continuous (or even smoother) functions, popular in the mathematical literature, is not so appropriate in particle physics phenomenology, where experimental cuts usually imply discontinuous steps in complicated shapes.

Even more relevant for our purposes than the Green's functions themselves are the *connected Green's functions* $c_n(x_1, x_2, \ldots, x_n)$, which form the irreducible building blocks of the Green's functions. We have

$$\begin{aligned} g_0() &= 1 \quad , \\ g_1(x_1) &= c_1(x_1) \quad , \\ g_2(x_1, x_2) &= c_1(x_1)c_1(x_2) + c_2(x_1, x_2) \quad , \\ g_3(x_1, x_2, x_3) &= c_1(x_1)c_1(x_2)c_1(x_3) + c_1(x_1)c_2(x_2, x_3) + c_1(x_2)c_2(x_3, x_1) \\ &\quad + c_1(x_3)c_2(x_3, x_1) + c_3(x_1, x_2, x_3) \quad , \end{aligned} \tag{31}$$

and so on; the following recursive definition holds:

$$g_n(x_1, x_2, \ldots, x_n) = \sum_{\mathcal{P}} \sum_{m=1}^{n} \frac{c_m(x_1, z_2, z_3, \ldots, z_m)}{(m-1)!} \frac{g_{n-m}(z_{m+1}, z_{m+2}, \ldots, z_n)}{(n-m)!} \quad , \tag{32}$$

where $(z_2, z_3, \ldots, z_n)$ is a permutation $\mathcal{P}$ of $(x_2, x_3, \ldots, x_n)$: the first sum runs over all $(n-1)!$ such permutations. The connected Green's functions are, of course, also totally symmetric. The nice thing about connected Green's functions is that in many cases there is only a finite number of non-vanishing ones; for instance, for a Gaussian problem class such as the Wiener sheet measure, only c_2 is nonzero: in that case g_n is zero if n is odd, and g_{2n} consists of $(2n)!/2^n n!$ terms, each a product of n factors c_2.

4.3 Induced discrepancy

We now proceed to derive how the set of Green's functions induces a quantity with the general properties of a discrepancy. If the integrand is chosen 'at

random' from the problem class, the integration error η will again be random even for a fixed point set, and we can derive its expectation value and moments. Denoting averages over the problem class by brackets we have

$$\begin{aligned} \langle\, \eta^m \,\rangle_f &= \langle\, (S-J)^m \,\rangle_f = \sum_{p=0}^{m} \left(\begin{array}{c} m \\ p \end{array} \right) (-1)^{m-p} M_{m,p} \quad , \\ M_{m,p} &= \frac{1}{N^p} \sum_{k_1,k_2,\ldots,k_p} \int\limits_K d\mu(z_{p+1}) \int\limits_K d\mu(z_{p+2}) \cdots \\ &\quad \cdots \int\limits_K d\mu(z_m)\; g_m(x_{k_1}, x_{k_2}, \ldots, x_{k_p}, z_{p+1}, z_{p+2}, \ldots, z_m) \quad . \end{aligned} \tag{33}$$

Now, we express the g_m in the connected Green's functions. The combinatorial factors are quite involved, but the result can be summarized as follows. The moment $\langle\, \eta^m \,\rangle$ consists of a sum of all powers of all possible combinations of c's, with each c having as arguments all possible combinations of summed-over point set members and integrated-over points in K. Let us define

$$\begin{aligned} a_{n,k} &= \frac{1}{N^k} \sum_{k_1,k_2,\ldots,k_k} \int\limits_K d\mu(z_{k+1}) \int\limits_K d\mu(z_{k+2}) \cdots \\ &\quad \cdots \int\limits_K d\mu(z_n)\; c_n(x_{k_1}, x_{k_2}, \ldots, x_{k_k}, z_{k+1}, z_{k+2}, \ldots, z_n) \quad . \end{aligned} \tag{34}$$

The various $a_{n,k}$ with the same n will always occur in the combination

$$d_n = \sum_{k=0}^{n} \frac{(-1)^{n-k} a_{n,k}}{k!(n-k)!} \quad , \tag{35}$$

and the m^{th} moment of η is given by

$$\langle\, \eta^m \,\rangle_f = m! \sum_{\{\nu_n\}} \frac{d_1^{\nu_1}}{\nu_1!} \frac{d_2^{\nu_2}}{\nu_2!} \frac{d_3^{\nu_3}}{\nu_3!} \cdots \quad , \tag{36}$$

where the sets of integers $\{\nu_n\}$ are restricted by $\nu_n \geq 0$, and $\nu_1 + 2\nu_2 + 3\nu_3 + \cdots = m$. Note that, whenever c_n is a constant, d_n is zero.

We may now introduce the moment-generating function for η:

$$\langle\, e^{z\eta} \,\rangle_f = \sum_{m \geq 0} \frac{z^m}{m!} \langle\, \eta^m \,\rangle_f = \exp\left(\sum_{m \geq 1} z^m d_m \right) \quad . \tag{37}$$

In many cases, we may be able to construct the actual probability distribution $P(\eta)$ for η by inverse Mellin transform. We are assured, in fact, that this transform always exists – after all, if the integrands have *some* probability

distribution over the problem class, then the integration error also has *some* distribution.

More important is the realization that, if the problem class is Gaussian in the sense that the only nonzero connected Green's function is c_2, then $P(\eta)$ is necessarily a normal distribution, and we only have to know $\langle\ \eta^2\ \rangle$ to know all confidence levels. In addition, note that whenever d_1 is nonzero, the integration will be biased. If, in addition to d_2, also d_4 is nonzero, it had better be negative – but, if it is positive, the problem class itself is not well-defined.

What, now, is the use of all this for practical error estimates? This lies in the following *factorization property:* suppose that we can find a function $h(x;y)$, with x in K, and y being *some* parameter in *some* space L, such that the connected Green's function can be expressed in terms of h as follows:

$$c_n(x_1, x_2, \ldots, x_n) = \int\limits_L dy\ h(x_1;y)h(x_2;y)\cdots h(x_n;y) \quad : \tag{38}$$

then, we can easily derive that d_n has the simple form

$$\begin{aligned} d_n &= \frac{1}{n!}\int\limits_L dy\ H(y)^n \quad , \\ H(y) &= \frac{1}{N}\sum_k h(x_k;y) - \int\limits_K d\mu(x)h(x;y) \quad . \end{aligned} \tag{39}$$

The function $H(y)$ is the induced local discrepancy, and d_n is a measure of induced global discrepancy. $H(y)$ has precisely the form of Eq.(11): it measures how well the function $h(x;y)$ is integrated by the point set, for a given value of y: but, note that $h(x;y)$ itself is not necessarily a member of the problem class. As we shall see, many problem classes have the factorization property. For such a problem class, we then have the following strategy available:

1. determine the Green's functions;
2. determine the connected Green's function c_2;
3. find a parameter y and a function $h(x,y)$ such that factorization holds;
4. for a given point set, determine $H(y)$ and d_2;
5. we have then estimated the expected moment $\langle\ \eta^2\ \rangle$ for the integration error if we use this point set on an integrand picked at random from the problem set.

4.4 Orthonormal-function measures

We shall now discuss how to apply the above formalism for a special kind of problem class. As usual, we start with the one-dimensional case. Let us take

$K = [0,1]$, and $d\mu(x) = dx$, the usual uniform measure. Furthermore, we define a set of functions $u_n(x)$, $n = 0, 1, 2, 3, \ldots$, that are *orthonormal*:

$$\int_0^1 dx\ u_m(x)u_n(x) = \delta_{m,n} \quad , \quad u_0(x) = 1. \tag{40}$$

As an example, we may take the u_m to be orthonormal polynomials, and we shall presently discuss some other examples. We then define our problem class to consist of all functions that can be written as a linear combination of these orthonormal ones:

$$f(x) = \sum_{n\geq 0} v_n u_n(x) \quad , \tag{41}$$

so that the coefficients v_n completely determine $f(x)$. If our orthonormal set is even *complete,* that is, if

$$\lim_{N\to\infty} \sum_{n=1}^{N} u_n(x_1)u_n(x_2) = \delta(x_1 - x_2) \quad , \tag{42}$$

we can actually approximate any $f(x)$ to an arbitrary accuracy by members of the problem class.

A measure on the problem class can now be specified by giving the probability density of each individual coefficient v_n. For simplicity, we take these distributions to be normal, and we can write

$$\mathcal{D}f = \prod_{n\geq 0} dv_n \frac{1}{\sqrt{2\pi\sigma_n^2}} \exp\left(-\frac{v_n^2}{2\sigma_n^2}\right) \quad , \tag{43}$$

where σ_n is the width, which may be different for different v_n. Now, the only nonzero connected Green's function is c_2, and we always have the factorization property:

$$\begin{aligned} c_2(x_1, x_2) &= \sum_{n\geq 0} \sigma_n^2 u_n(x_1)u_n(x_2) \quad = \quad \int_0^1 dy\ h(x_1; y)h(x_2; y) \quad , \\ h(x; y) &= \sum_{n\geq 0} \sigma_n u_n(x)u_n(y) \quad . \end{aligned} \tag{44}$$

Thus, we can immediately write down the induced discrepancy:

$$D_2^{\text{ortho}} = \frac{1}{N^2} \hat{\sum_n} \sigma_n^2 \sum_{k,l} u_n(x_k)u_n(x_l) \quad , \tag{45}$$

where the caret denotes a sum over all n *except* $n = 0$: this is due to the fact that $u_0(x) = 1$ is a constant, and obviously the constant component of an integrand is always estimated with zero error by any point set.[2]

[2]The quadratic discrepancy D_2^{ortho} is defined here as $2d_2$, such that it is the expectation value of $\left\langle\ \eta^2\ \right\rangle_f$.

The extension to more dimensions is fairly straightforward, if we replace the simple one-dimensional enumeration $n = 0, 1, 2, \ldots$ by a multidimensional one:

$$n = \vec{n} = (n_1, n_2, \ldots, n_s) \quad , \quad n^\mu = 0, 1, 2, 3, \ldots \quad . \tag{46}$$

Then, we simply put

$$u_{\vec{n}}(x) = \prod_\mu u_{n^\mu}(x^\mu) \quad , \tag{47}$$

and (for a Gaussian measure as above) we again have

$$D_2^{\text{ortho}} = \frac{1}{N^2} \hat{\sum_{\vec{n}}} \sigma_{\vec{n}}^2 \sum_{k,l} u_{\vec{n}}(x_k) u_{\vec{n}}(x_l) \quad , \tag{48}$$

where now the sum runs over all vectors $\vec{n} = (n_1, \ldots, n_s)$ except the null vector $(0, \ldots, 0)$.

In the following we shall discuss some explicit examples of these problem classes, and the particular form of their induced discrepancies. At this point, however, we may already note that, if the orthonormal set is complete, and if σ_n does not go to zero as $n \to \infty$, then D_2^{ortho} will be infinite, since it will have contributions proportional to $\delta(0)$. In concrete terms, in such a problem class the typical integrands show such wild behaviour that the average squared integration error is infinite. In our explicit examples we shall see how the behaviour of σ_n relates to the smoothness properties (or lack thereof) of the functions in the problem class.

4.5 Examples

We shall now discuss some explicit examples of problem classes, and their induced quadratic discrepancies. The first two examples are not of the orthonormal-function type; the last two are.

4.5.1 Wiener sheet measure: the Woźniakowski Lemma

One of the more popular problem class measures is the Wiener measure. It is widely used in field theory and statistical physics, and describes, basically, a Brownian motion. Although its precise definition, especially in more dimensions, is subtle, we can describe it in the following manner. Let us start with the case $s = 1$, and 'discretize' the unit interval $[0, 1)$ into many small subintervals (not necessarily equal), to wit $[0, x_1)$, $[x_1, x_2)$, $[x_2, x_3)$, and so on. The value of the function jump from x_p to x_{p+1} is then assumed to be distributed normally around zero, with variance given by the interval size (not its square!). In this way, the Wiener measure becomes transitive, in that the distribution for $f(x_{p+2})$, when we jump to x_{p+2} directly from x_p, is the same as that obtained when we first jump from x_p to x_{p+1}, and then from x_{p+1} to x_{p+2}. Therefore, we may insert new interval endpoints in-between the old ones at will, and so

approach the continuum limit. The *derivative* of the function will then have the Green's function

$$\langle\, f'(x_1)f'(x_2)\,\rangle_f = \delta(x_1 - x_2) \quad . \tag{49}$$

This singular behaviour comes from the fact that the average value of the jump in f is proportional to the *square root* of the jump size, whereas for differentiable functions it would, asymptotically, be proportional to the jump size itself. The Wiener measure is, therefore, completely dominated by functions that are everywhere continuous but nowhere differentiable.

By integration, we can simply derive the Green's function for the function values themselves:

$$\langle\, f(x_1)f(x_2)\,\rangle_f = \min(x_1, x_2) \quad . \tag{50}$$

In more dimensions, we have the *Wiener sheet measure,* $K = I_s$, the rôle of the derivative $f'(x)$ is played by the multiple derivative:

$$f_s(x) \equiv (d^s/dx_1 dx_2 \cdots dx_s) f(x) \quad , \tag{51}$$

and we have

$$\begin{aligned} \langle\, f_s(x_1)f_s(x_2)\,\rangle_f &= \prod_\mu \delta(x_1^\mu - x_2^\mu) \quad , \\ \langle\, f(x_1)f(x_2)\,\rangle_f &= \prod_\mu \min(x_1^\mu, x_2^\mu) \quad . \end{aligned} \tag{52}$$

In addition, we have $\langle\, f(x)\,\rangle = 0$, so that the only nonzero connected Green's function is

$$\begin{aligned} c_2(x_1, x_2) &= \prod_\mu \min(x_1^\mu, x_2^\mu) = \int_K dy\; h(x_1; y) h(x_2; y) \quad , \\ h(x; y) &= \prod_\mu \theta(x^\mu - y^\mu) \quad , \end{aligned} \tag{53}$$

where we have also indicated the factorization. Inspection tells us that, in fact, $h(x; y)$ is equal to the function χ discussed for the classical discrepancy, only with the points x and y reflected: $x^\mu \to 1 - x^\mu$, $y^\mu \to 1 - y^\mu$. We therefore arrive immediately at the so-called *Woźniakowski lemma: for integrands distributed according to the Wiener sheet measure, the average squared integration error is equal to the classical D_2 discrepancy, for the reflected point set.* For random points, the expected value of the quadratic induced discrepancy is

$$D_2^{\text{Wiener}} = \frac{1}{N}\left(\frac{1}{2^s} - \frac{1}{3^s}\right) \quad , \tag{54}$$

as discussed above. The reflection of the point set is in practice not very important since any self-respecting point set is, to a high degree, more or less invariant under reflection.

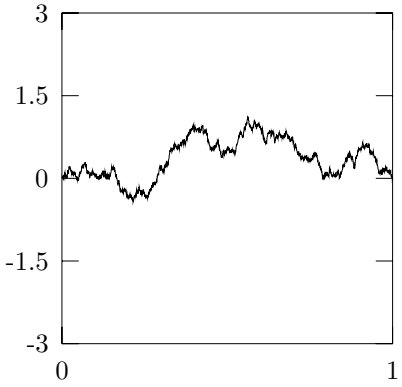


Figure 3: A typical integrand under the Wiener measure.

This result, published in [11], was the first instance of the relation between problem classes and discrepancies. It is seen here to arise as a special case of a more generally applicable induction of discrepancies. Since the Wiener measure as discussed is Gaussian, we have also established that the error will, in fact, be normally distributed over the problem class, which fact was not noticed in [11].

4.5.2 Folded Wiener sheet measure

As mentioned, integrands that are 'typical' under the Wiener measure may not be of the kind studied in particle physics phenomenology: as an illustration, we present, in fig.6, a 'typical' integrand in one dimension. Although they are continuous, such integrands have no additional smoothness properties.

As an approximation to smoother functions, one may consider so-called *folded Wiener sheet measures,* which are obtained by (repeated) integration, so that the fractal zig-zag structure becomes hidden in some higher derivative[12]. A typical integrand under the n times folded Wiener sheet measure is defined, recursively, by

$$f^{(n)}(x) = \int_0^{x_1} dx'_1 \int_0^{x_2} dx'_2 \cdots \int_0^{x_s} dx'_s f^{(n-1)}(x') \quad , \tag{55}$$

where $f^{(0)}(x)$ is the integrand chosen from the Wiener measure. For these folded measures, the induced discrepancy can therefore be obtained trivially by repeated integration of h in Eq.(53) over the variables x:

$$h^{(n)}(x;y) = \prod_\mu \frac{(x^\mu - y^\mu)^n}{n!} \theta(x^\mu - y^\mu) \quad , \tag{56}$$

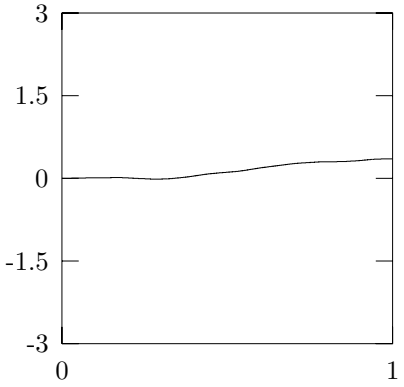


Figure 4: A typical integrand under the one time folded Wiener measure.

and, for random points, the expected quadratic induced discrepancy in this case is given by

$$D_2^{\text{folded Wiener}} \quad = \quad \frac{1}{N}\left(\frac{1}{n!}\right)^{2s}\left(\prod_{\mu}\frac{1}{(2n+1)(2n+2)} - \prod_{\mu}\frac{1}{(n+1)^2(2n+3)}\right) \quad . \tag{57}$$

This discrepancy, and hence the expected integration error, decreases extremely rapidly with increasing degree of folding. This can be understood easily if we plot, say, the one times folded equivalent of fig.6, which is given in fig.7.

One sees that, even under only one folding, we obtain functions that are extremely smooth, and it is not surprising that such integrands can be treated with very small error. We conclude that, although the Wiener measure may be applicable in some fields (the behaviour of the stock market, say), it is not really a valid model of typical integrands in particle physics phenomenology.

4.5.3 Fourier measure

We shall now discuss another problem class measure, using orthonormal functions as described above: again, first in one dimension. Let us take

$$u_0(x) = 1 \ , \ \ u_{2n}(x) = \sqrt{2}\cos 2\pi n x \ , \ \ u_{2n-1}(x) = \sqrt{2}\sin 2\pi n x \ . \tag{58}$$

These functions are even a complete set: if we disregard the Gibb's phenomenon, which we may for purposes of integration, every practically relevant function can be approximated arbitrarily closely by a combination of u_n's. In appendix B we present a short proof for completeness. Also, in many physical cases the Gaussian distribution of the magnitude of the several higher modes is quite reasonable.

We may immediately write down the form of the induced quadratic discrepancy, as given in Eq.(45). At this point, we may make another assumption. The two functions u_{2n} and u_{2n-1} describe the mode with frequency n, with two phases apart by $\pi/2$. These may be combined into a single mode, with a single phase x_0:

$$\begin{aligned} v_{2n-1}u_{2n-1}(x) + v_{2n}u_{2n}(x) &= \sqrt{2}v_{2n-1}\sin 2\pi nx + \sqrt{2}v_{2n}\cos 2\pi nx \\ &\propto \cos 2\pi n(x+x_0) \quad . \end{aligned} \tag{59}$$

Now, if we, reasonably, assume that x_0 is uniformly distributed between $-\pi/2$ and $\pi/2$, we can infer that this necessitates

$$\sigma_{2n-1} = \sigma_{2n} \quad , \quad n = 1, 2, \ldots \quad , \tag{60}$$

and in the following we shall use this simplification. The discrepancy then takes the form

$$\begin{aligned} D_2^{\text{Fourier}} &= \frac{2}{N^2}\sum_{n>0}\sigma_{2n}^2 \sum_{k,l}\cos 2\pi n(x_k - x_l) \\ &= \frac{2}{N^2}\sum_{n>0}\sigma_{2n}^2 |\sum_k \exp(2i\pi n x_k)|^2 \quad . \end{aligned} \tag{61}$$

Note that we have two equivalent expressions here, that are however of different consequence in computer implementation. The first line contains a double sum over the point set, and it can be evaluated in time $\mathcal{O}(N^2)$, if we know a closed expression for $\sum \sigma_{2n}^2 \cos(2\pi nx)$. For instance, we may assume $\sigma_{2n} = 1/n$, in which case we have

$$\sum_{n\geq 1}\frac{1}{n^2}\cos(2\pi nx) = \frac{\pi^2}{6}\left(1 - 6x(1-x)\right) \quad . \tag{62}$$

On the other hand, we may use a single sum as in the second line of Eq.(61); but then, we have to sum over the modes explicitly. Supposing that we only really need the modes up to $n = m$, the discrepancy can then be computed in time $\mathcal{O}(mN)$.

The multidimensional extension is again rather trivial, if we assume the orthonormal functions as a direct product of the one-dimensional ones, as discussed above. We may immediately write out the induced quadratic discrepancy. With a little algebra, we may, in fact, arrive at

$$\begin{aligned} D_2^{\text{Fourier}} &= \frac{1}{N^2}\hat{\sum_{\vec{n}}}\tau_{\vec{n}} |\sum_k \exp(2i\pi\vec{n}\cdot\vec{x}_k)|^2 \quad , \\ \tau_{(n^1,n^2,\ldots,n^s)} &= \sigma^2_{(2|n^1|,2|n^2|,\ldots,2|n^s|)} \quad , \end{aligned} \tag{63}$$

where the sum over $\vec{n}$ now runs over *all* integer values (both negative and positive) of the components, excluding the null vector $(0, 0, \ldots, 0)$.

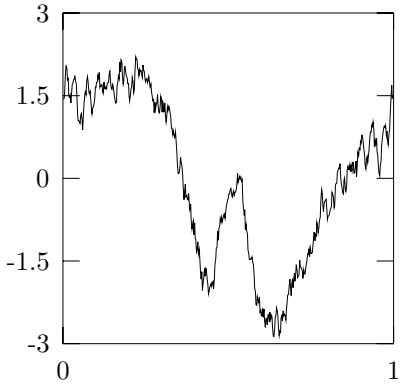


Figure 5: A typical integrand under the Fourier measure, with $\sigma_{2n-1} = \sigma_{2n} = 1/n$.

We finish this section by pointing out a relation between our results for the induced discrepancy and the exponential sums discussed at length in [6, 7] (for instance, see [6], chapter 2, section 2). There, the classical Kolmogorov discrepancy is related to exactly the same kind of exponential, the only difference being the absence of the factor $\tau_{\vec{n}}$. What does this imply? Note that, in one dimension (for simplicity) we have

$$\begin{aligned}
\left\langle \int_0^1 dx f(x)^2 \right\rangle_f &= \sum_{n\geq 0} \sigma_n^2 \quad , \\
\left\langle \int_0^1 dx f'(x)^2 \right\rangle_f &= 4\pi^2 \sum_{n\geq 0} n^2 \sigma_{2n}^2 \quad , \\
\left\langle \int_0^1 dx f''(x)^2 \right\rangle_f &= 16\pi^4 \sum_{n\geq 0} n^4 \sigma_{2n}^2 \quad , \qquad (64)
\end{aligned}$$

so that the convergence of these sums informs us about the smoothness of the typical integrands: the smoother, on average, the integrand, the faster the σ_n has to approach zero. If $\sigma_n = 1$, as implicit in the treatment as given in [6], the average integrand is not even quadratically integrable! An example of such an integrand is given in fig.8. From the point of view of the present paper, choosing a slowly decreasing $\sigma_{\vec{n}}$ appears to lead to a certain kind of overkill in the selection of good point sets.

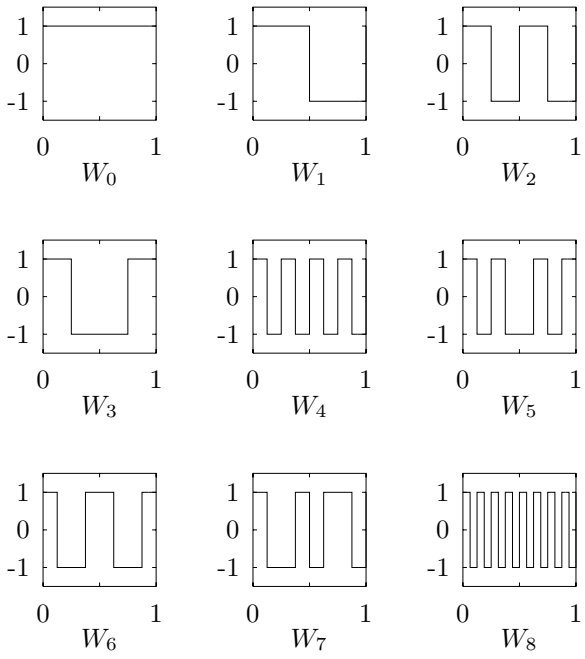


Figure 6: The first nine Walsh functions, $W_0(x)$ to $W_8(x)$.

4.5.4 Walsh measure

Now we shall discuss yet another complete set of orthonormal functions, the so-called *Walsh* functions. Again, we start with $s = 1$. We first introduce the Rademacher functions $\phi_n(x)$: these are periodic with period one, and, for $0 \leq x < 1$,

$$\phi_1(x) = \begin{cases} 1 & \text{if } x < 1/2 \\ \text{-1} & \text{if } x \geq 1/2 \end{cases} \quad , \quad \phi_n(x) = \phi_{n-1}(2x) \quad . \tag{65}$$

The Walsh functions $W_n(x)$ are now given as follows: let the binary decomposition of n be

$$n = n_1 + 2n_2 + 2^2 n_3 + 2^3 n_4 + \cdots \quad , \tag{66}$$

then $W_n(x)$ is

$$W_n(x) = \phi_1(x)^{n_1} \phi_2(x)^{n_2} \phi_3(x)^{n_3} \cdots \quad , \tag{67}$$

and we define $W_0(x) \equiv 1$. As an example, we present the first few Walsh functions in fig.9. The Walsh functions form a complete set, and we may use them to construct a problem class, defined under the *Walsh measure*. In fig.10, we show a typical integrand under the Walsh measure, with $\sigma_n = 1/n$.

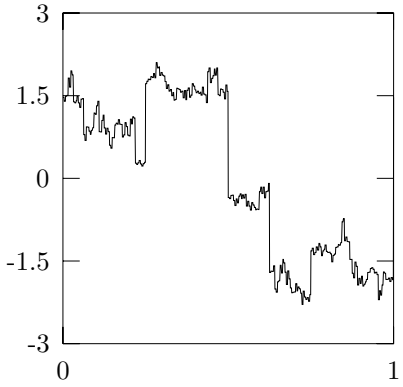


Figure 7: A typical integrand under the Walsh measure, using W_n with $n = 0, 1, \ldots, 200$, and $\sigma_n = 1/n$.

Some insight in the computational character of the Walsh functions can be gained as follows. For any x in the unit interval $[0, 1)$, let us define its binary notation by

$$x = x_1 2^{-1} + x_2 2^{-2} + x_3 2^{-3} + \cdots \quad ; \tag{68}$$

it is, then, easy to see that

$$\phi_n(x) = (-1)^{x_n} \quad , \tag{69}$$

and therefore, the Walsh functions just measure certain parities: with n decomposed as above, we have

$$W_n(x) = (-1)^{n_1 x_1 + n_2 x_2 + n_3 x_3 + \cdots} \quad . \tag{70}$$

Clearly, in a language with command of bitwise operations, such an evaluation can easily be implemented. Note, moreover, that as long as we restrict the range of n, our assumption of the real-number model of computation is justified: if the least significant bit of x has a value of 2^{-p}, then the finite word length of our computer is irrelevant as long as $n < 2^p$.

The extension to the case of more dimensions is of course straightforward, but we shall not discuss this further.

5 Discrepancies for random point sets

5.1 Introduction: the need for comparison

So far, we have studied how the choice of integration problem classes relates to discrepancy-like properties of the integration point set. Obviously, we should like to improve on classical Monte Carlo as much as we can. This raises the question

we shall study in the rest of this paper: suppose we have a point set (or sequence) at our disposal, which we want to use for integration. Suppose, moreover, that we have computed its extreme, or another induced, discrepancy. To what extent is this discrepancy better than what we could expect for random points? To answer this question, we have to compute, for a given value of the discrepancy, the probability that purely random points would give this value (or less) for the discrepancy. That means, we have to know the probability distribution of the discrepancy, which is now viewed as a random quantity through its dependence on the set of random points. Of course, we could restrict ourselves to the relatively simple question of the average discrepancy for random points, but it is much better to know the confidence levels, and, moreover, we shall see that they can actually be computed in many cases. Another point is that we might use a source of (pseudo)random numbers to produce a histogram of the discrepancy for a number of generated point sets — but in practice this is not feasible, since we would, before generating such a histogram, have to ascertain that the produced sequence is actually acceptably random: and how to do this without comparing to the theoretical expectations?

5.2 Moment-generating function

We shall study the moment-generating function of the linear and quadratic induced discrepancies, that is: if for a set of N truly random points, the linear discrepancy equals x, we then try to find the form of

$$G_1(z) = \langle \exp(zx) \rangle \quad , \tag{71}$$

and then, we can (numerically) find the probability density for x:

$$f_1(x) = \frac{1}{2\pi} \int_{-\infty}^{\infty} e^{-izx} G_1(iz) \quad ; \tag{72}$$

and, similarly, we can find G_2 and f_2 for the quadratic discrepancy. Although the linear discrepancy is not as important as the quadratic one, we shall discuss it first since it indicates how the limit of large N can simply be taken.

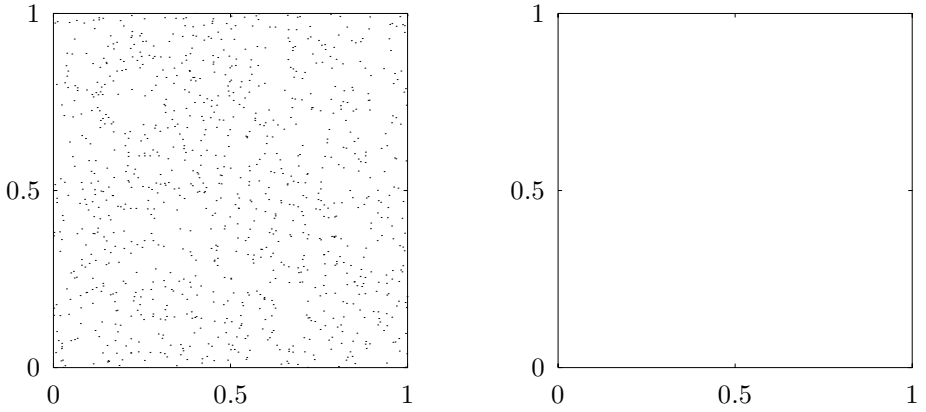
5.2.1 Linear Discrepancy

Let us assume that, for a given problem class, the discrepancy function $h(x; y)$ is known. The linear discrepancy is then given by

$$D_1 = \frac{1}{N} \int_L dy \sum_k \omega_k(y) \ , \ \omega_k(y) = h(x_k; y) - \int_K d\mu(x)\, h(x; y) \quad . \tag{73}$$

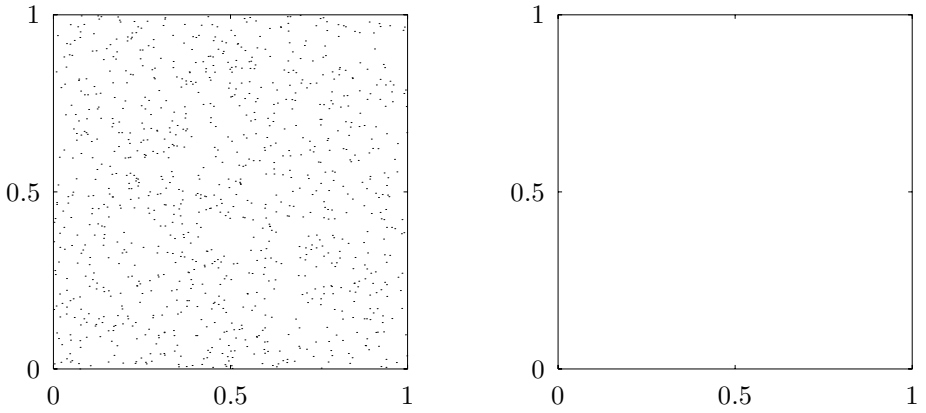
To get the moment-generating function, we have to evaluate

$$\langle D_1^p \rangle = \frac{1}{N^p} \int_L dy_1 \int_L dy_2 \cdots \int_L dy_p \left\langle \sum_{k_{1,2,\ldots,p}} \omega_{k_1}(y_1)\omega_{k_2}(y_2)\cdots\omega_{k_p}(y_p) \right\rangle \quad , \tag{74}$$



(a) 1000 points

(b) 40000 points



(c) 1000 points

(d) 40000 points

Figure 8: Some points generated in two dimensions using a good pseudorandom generator (RANLUX, level 4 [8]).

for all $p \geq 0$. Let us consider the sum over the indices k in detail. All indices run from 1 to N, and some of the indices may be equal to others. Those contributions where the indices fall into m distinct values (that is, some of the k_i all have one value, some other all have another value, a third group has another value, and so on), enter with a combinatorial factor $N(N-1)(N-2)\cdots(N-m+1)$. It is clear that the largest combinatorial factor is obtained if as many as possible of the indices k are different. Now, we have

$$\langle\ w_k(y)\ \rangle = 0 \quad , \tag{75}$$

and the largest nonzero contribution is therefore the one coming from the cases where the indices come in pairs. This also means that $\langle\ D_1^p\ \rangle$ vanishes (to leading order in N) if m is odd, and

$$\langle\ D_1^{2m}\ \rangle \sim \frac{1}{N^{2m}} N^m \frac{(2m)!}{2^m m!} \left(\int_L dy_1 dy_2\ \langle\ w_k(y_1) w_k(y_2)\ \rangle \right)^m \quad , \tag{76}$$

where we have also made the approximation

$$N(N-1)(N-2)\cdots(N-m+1) \sim N^m \quad , \tag{77}$$

which is justified when $N \gg m^2$. The moment-generating function can now be written as

$$\begin{aligned} \langle\ e^{zD_1}\ \rangle &= \sum_{m\geq 0} \frac{z^{2m}}{(2m)!} \langle\ D_1^{2m}\ \rangle = \exp\left(\frac{z^2}{2N} W^n\right) \quad , \\ W &= \int_K d\mu(x) \int_L dy_1 dy_2\ h(x;y_1) h(x;y_2) \\ &\quad - \left(\int_K d\mu(x) \int_L dy\ h(x;y) \right)^2 \quad , \end{aligned} \tag{78}$$

and the probability density can be obtained by Fourier transform. In the large-N limit, the value of D_1 has a Gaussian distribution with mean zero and width $\sqrt{W/N}$. Of course, all this is nothing but the Central Limit Theorem, but we shall use the above combinatorial arguments also in our study of the quadratic discrepancy.

The value of W depends, of course, on the problem class. The Wiener measure discussed above leads, in one dimension, to

$$\begin{aligned} W &= \int_0^1 dx dy_1 dy_2\ \theta(y_1 - x)\theta(y_2 - x) - \left(\int_0^1 dx dy\ \theta(y - x) \right)^2 \\ &= \int_0^1 dy_1 dy_2 \min(y_1, y_2) - \left(\int_0^1 dy\ y \right)^2 \\ &= \frac{1}{3} - \frac{1}{4} \quad , \end{aligned} \tag{79}$$

and in higher dimension, we obviously have

$$W = \left(\frac{1}{3}\right)^s - \left(\frac{1}{4}\right)^s \quad . \tag{80}$$

In the case of problem classes defined using orthonormal functions as above, we notice that

$$\int\limits_L dy\ h(x;y) = \sum_{n>0} \sigma_n u_n(x) \int\limits_L dy\ u_n(y) = 0 \quad . \tag{81}$$

and therefore D_1 is always strictly zero for such problem classes.

5.2.2 Quadratic Discrepancy

We now turn to the somewhat more complicated case of the quadratic induced discrepancy. In order to avoid factors of N cropping up everywhere, we shall consider ND_2 rather than D_2 itself. We have

$$D_2 = \frac{1}{N^2} \int\limits_L dy\ \sum_{k,l} \omega_k(y)\omega_l(y) \quad . \tag{82}$$

Its m^{th} moment is given by

$$\begin{aligned} \langle\ N^m D_2^m\ \rangle \quad &= \quad \frac{1}{N^m} \int\limits_L dy_1 \int\limits_L dy_2 \cdots \int\limits_L dy_m \sum_{k_{1,2,\ldots,m}} \sum_{l_{1,2,\ldots,m}} \\ &\langle\ \omega_{k_1}(y_1)\omega_{l_1}(y_1)\omega_{k_2}(y_2)\omega_{l_2}(y_2)\cdots\omega_{k_m}(y_m)\omega_{l_m}(y_m)\ \rangle \end{aligned} \tag{83}$$

By the same combinatorial argument as before, in the large-N limit we only have to consider the contribution to the sum that consists only of pairs of ω's:

$$\begin{aligned} \alpha(y_1,y_2) \quad &\equiv \quad \langle\ \omega_k(y_1)\omega_k(y_2)\ \rangle \quad = \quad \beta(y_1,y_2) - v(y_1)v(y_2) \quad , \\ \beta(y_1,y_2) \quad &= \quad \int\limits_K dx\ h(x;y_1)h(x;y_2) \quad , \\ v(y) \quad &= \quad \int\limits_K dx\ h(x;y) \quad . \end{aligned} \tag{84}$$

At this point, a few remarks are in order. In the first place, since every parameter y_i occurs twice, the various pairs cannot be considered independently: rather, we have to deal with objects of the form

$$Z_n = \int\limits_L dy_1\ dy_2 \cdots dy_n \alpha(y_1,y_2)\alpha(y_2,y_3)\cdots\alpha(y_{n-1},y_n)\alpha(y_n,y_1) \quad . \tag{85}$$

Another consideration is that of factorization in more dimensions. Whereas, in higher dimensions, the functions $h(x;y)$ themselves usually factorize into a

product of one-dimensional factors (as for the Wiener and orthonormal problem classes), this no longer holds for the functions α: only the functions β and v factorize. We therefore decide to build up $\langle\, D_2^m \,\rangle$ from two distinct objects:

$$\begin{aligned} C_n &= \int\limits_L dy_1\, dy_2 \cdots dy_n \beta(y_1,y_2)\beta(y_2,y_3)\cdots\beta(y_{n-1},y_n)\beta(y_n,y_1) \quad , \\ O_n &= \int\limits_L dy_1\, dy_2 \cdots dy_n v(y_1)\beta(y_1,y_2)\cdots\beta(y_{n-1},y_n)v(y_n) \quad , \end{aligned} \tag{86}$$

which we call *closed strings* and *open strings,* respectively. We can then concentrate on the one-dimensional case first, and in more dimensions simply use

$$\begin{aligned} C_n(s\text{-dim}) &= C_n(1\text{-dim})^s \quad , \\ O_n(s\text{-dim}) &= O_n(1\text{-dim})^s \quad . \end{aligned} \tag{87}$$

Keeping careful account of the combinatorial factors, we have

$$\begin{aligned} \langle\, N^m D_2^m \,\rangle &= \sum_{p_{1,2,\ldots}} \sum_{q_{1,2,\ldots}} \frac{m!}{p_1!p_2!\cdots q_1!q_2!\cdots} \frac{(2r)!}{2^r r!} \\ &\qquad \times \prod_{l\geq 1} \left(\frac{2^{l-1}}{l} C_l\right)^{p_l} \prod_{n\geq 1} \left(-2^{n-1} O_n\right)^{q_n} \quad , \end{aligned} \tag{88}$$

with the constraints

$$\begin{aligned} m &= p_1 + 2p_2 + 3p_3 + \cdots + q_1 + 2q_2 + 3q_3 + \cdots \quad , \\ r &= q_1 + q_2 + q_3 + \cdots \quad . \end{aligned} \tag{89}$$

The moment-generating function is, therefore,

$$\begin{aligned} G_2(z) &\equiv \sum_{m\geq 0} \frac{z^m}{m!} \langle\, N^m D_2^m \,\rangle = \frac{\exp(\psi(z))}{\sqrt{\chi(z)}} \quad , \\ \psi(z) &= \sum_{n\geq 1} \frac{(2z)^n}{2n} C_n \quad , \\ \chi(z) &= 1 + \sum_{n\geq 1} (2z)^n O_n \quad . \end{aligned} \tag{90}$$

So, provided we can compute the closed and open strings for a given problem class, we can compute the probability density $f_2(x)$ for ND_2 to have the value x, by Fourier transform:

$$f_2(x) = \frac{1}{2\pi} \int\limits_{-\infty}^{\infty} dz\, e^{-izx} G_2(iz) = \frac{1}{\pi} \int\limits_0^{\infty} dz\, \mathrm{Re}\left(e^{-izx} G_2(iz)\right) \quad , \tag{91}$$

where we have used $G_2(z^*) = G_2(z)^*$, since $G_2(z)$ is real for z real and sufficiently small. Finally, by taking various derivatives of $G_2(z)$ at $z = 0$, we can compute the moments of the ND_2 distribution, and find

$$\begin{aligned} \text{mean} &= C_1 - O_1 \quad , \\ \text{variance} &= 2(C_2 - 2O_2 + O_1^2) \quad , \\ \text{skewness} &= \sqrt{8}\, \frac{C_3 - 3O_3 + 3O_2O_1 - O_1^3}{(C_2 - 2O_2 + O_1^2)^{3/2}} \quad , \\ \text{kurtosis} &= 12\, \frac{C_4 - 4O_4 + 4O_3O_1 + 2O_2^2 - 4O_2O_1^2 + O_1^4}{(C_2 - 2O_2 + O_1^2)^2} \quad . \end{aligned} \tag{92}$$

5.3 The Wiener case

We shall now concentrate on the Wiener problem class. As usual we shall start with $s = 1$. In this case, we have

$$\beta(y_1, y_2) = \min(y_1, y_2) \quad , \quad v(y) = y \quad . \tag{93}$$

The strings can be computed most easily by first defining

$$A_n(x, y) = \int_0^1 dy_2\; dy_3 \cdots dy_n \beta(x, y_2)\beta(y_2, y_3) \cdots \beta(y_{n-1}, y_n)\beta(y_n, y) \quad , \tag{94}$$

so that

$$C_n = \int_0^1 dx\; A_n(x, x) \;\; , \;\; O_n = A_{n+1}(1, 1) \quad , \tag{95}$$

where the second line holds only in this, the Wiener, case. The objects A_n obey a recursion relation:

$$\begin{aligned} A_1(x, y) &= \beta(x, y) \quad , \\ A_n(x, y) &= \int_0^1 dz\; A_{n-1}(x, z)\beta(z, y) \;\; , \;\; n = 2, 3, \ldots \quad . \end{aligned} \tag{96}$$

They can most easily be computed using a generating function that itself satisfies the obvious integral equation:

$$F(t; x, y) = \sum_{n \geq 1} A_n(x, y)t^n \;\; , \;\; = t \int_0^1 dz\; F(t; x, z)\beta(z, y) + t\beta(x, y) \quad , \tag{97}$$

from which we shall establish a differential equation. Since

$$\min(x, 0) = 0 \;\; , \;\; \frac{\partial}{\partial y} \min(x, y) = \theta(x - y) \;\; , \;\; \frac{\partial^2}{\partial y^2} \min(x, y) = -\delta(x - y) \quad , \tag{98}$$

we can conclude that

$$\frac{\partial^2}{\partial y^2}F(t;x,y) = -tF(t;x,y) - t\delta(x-y) \quad , \tag{99}$$

with the boundary conditions

$$F(t;x,0) = 0 \ , \ \left[\frac{\partial}{\partial y}F(t;x,y)\right]_{y=0} = t\int\limits_0^1 dz\ F(t;x,z) + t \quad . \tag{100}$$

We can solve Eq.(99) by standard methods, and, with $t = u^2$, we have

$$F(t;x,y) = -u\theta(y-x)\sin(uy-ux) + \frac{u}{\cos u}\cos(u-ux)\sin(uy) \quad , \tag{101}$$

which is actually symmetric in x and y, as it should be. From Eq.(101) we can compute C_n and O_n in a straightforward manner, since

$$\sum_{n\geq 1} C_n t^n = \int\limits_0^1 dx\ F(t;x,x) \ , \ \sum_{n\geq 1} O_n t^n = \frac{1}{t}F(t;1,1) \quad , \tag{102}$$

and we arrive at

$$\begin{aligned} C_n &= \left(\frac{4}{\pi^2}\right)^n \xi(2n) \ = \ \frac{2^{2n-1}\left(2^{2n}-1\right)}{(2n)!}|B_{2n}| \quad , \\ O_{n-1} &= 2C_n \quad , \\ \xi(p) &= \sum_{n\geq 1}\frac{1}{(2n-1)^p} = (1-2^{-p})\zeta(p) \quad , \end{aligned} \tag{103}$$

where $\zeta(2n)$ is the Riemann zeta function, and B_{2n} is the Bernoulli number. The first few instances are

$$C_1 = \frac{1}{2} \quad , \quad C_2 = \frac{1}{6} \quad , \quad C_3 = \frac{1}{15} \quad , \quad C_4 = \frac{17}{630} \quad , \quad C_5 = \frac{31}{2835} \quad . \tag{104}$$

Incidentally, we also get $O_0 = 1$, in accordance with the fact that the series expansion of $\chi(z)$ must start with 1. The functions ψ and χ obtain the form

$$\begin{aligned} \psi(z) &= -\frac{1}{2}\sum_{n\geq 1}\log\left(1-\frac{8z}{\pi^2(2n-1)^2}\right) \ = \ -\frac{1}{2}\log\cos\sqrt{2z} \quad , \\ \chi(z) &= \sum_{n\geq 1}\frac{8}{\pi^2(2n-1)^2-8z} \ = \ \frac{1}{\sqrt{2z}}\tan\sqrt{2z} \quad . \end{aligned} \tag{105}$$

We can, using the series expansion forms, compute $\psi(iz)$ and $\chi(iz)$ to arbitrary accuracy. We might also use the fact that, in one dimension,

$$G_2(z) = \left[\frac{\sqrt{2z}}{\sin\sqrt{2z}}\right]^{1/2} \quad , \tag{106}$$

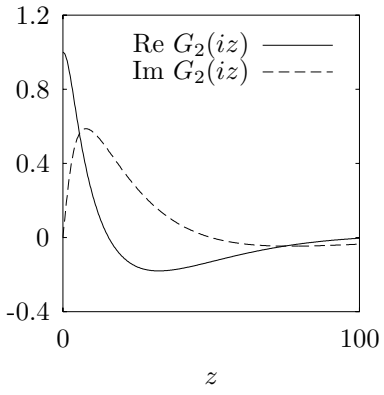


Figure 9: Real and imaginary part of $G_2(iz)$ for $s = 1$.

but, what with all the branch cuts this function has in the complex z plane, it is not easy to extend the computation of this more compact form so that $G_2(z)$ is adequately defined along the whole imaginary axis; and we shall use the form (105). Note that we may 'truncate' the series at large enough n, and sum the remainder in the approximation $z \ll n^2$: this gives, for instance

$$\begin{aligned} \psi(z) &\sim \frac{1}{2}z + \frac{1}{6}z^2 - \frac{1}{2}\sum_{n=1}^{K}\left[\log(1-z_n) + z_n + \frac{1}{2}z_n^2\right] \quad , \\ \chi(z) &\sim 1 + \frac{2}{3}z + \frac{1}{z}\sum_{n=1}^{K}\frac{z_n^3}{1-z_n} \quad , \\ z_n &\equiv \left(\frac{4}{\pi^2}\right)\frac{(2z)}{(2n-1)^2} \quad , \end{aligned} \tag{107}$$

where K is chosen large enough. In fig.11 we present the generating function $G_2(iz)$ for positive z.

The Fourier transform now gives us $f_2(ND_2)$ for $s = 1$. Its shape is given in fig.12, for $D_2 > 0$. We have checked that $f_2(x) = 0$ for negative x. The shape of the density is seen to be skewed, and not to approximate a Gaussian in the large-N limit (in fact, there is no Central Limit Theorem telling us that it should): the maximum is somewhat lower than the mean. It can be checked numerically that $f_2(x)$ is normalized to 1, and the first two moments are 1/6 and 1/20, respectively, in agreement with Eq.(92).

We now turn to the case of s dimensions. We have to replace $C_n \to C_n^s$, $O_n \to O_n^s$, and the nice analytic form (106) of the moment-generating function is lost. The series form for ψ and χ can be written in a form similar to that of Eq.(105), if we introduce $P_s(n)$, the *factor multiplicity* in s dimensions. This number counts in how many ways an odd integer n can be written as a product

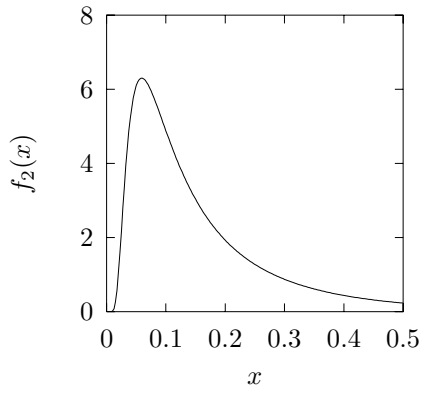


Figure 10: Probability density $f_2(ND_2)$, for $s = 1$.

of s odd integers, including factors 1. In appendix D we show how this can easily be computed. The functions ψ and χ then take the form

$$\begin{aligned}
\psi(z) &\sim \left(\frac{1}{2}\right)^s z + \left(\frac{1}{6}\right)^s z^2 - \frac{1}{2}\sum_{n=1}^{K} P_s(2n-1)\left[\log(1-z_n) + z_n + \frac{1}{2}z_n^2\right] \quad , \\
\chi(z) &\sim 1 + 2z\left(\frac{1}{3}\right)^s + \frac{2^s}{2z}\sum_{n=1}^{K} P_s(2n-1)\frac{z_n^3}{1-z_n} \quad , \\
z_n &\equiv \left(\frac{4}{\pi^2}\right)^s \frac{(2z)}{(2n-1)^2} \quad , \qquad (108)
\end{aligned}$$

We can now compute the $f_2(x)$ in these more-dimensional cases. In doing so, we must realize that $G_2(z)$ becomes increasingly flat, and hence $f_2(x)$ increasingly more peaked, with increasing s, as is evident from table 1, where we give the mean, standard deviation σ, and skewness of $f_2(x)$ as a function of s. Note that the deviation from a Gaussian, as indicated by the skewness, decreases with increasing s, but does so very slowly: the skewness decreases by a factor of 10 for an increase in s by about 113, and attains the value 0.001 only for $s = 389$. For 'medium' dimensionality, such as is common in high-energy phenomenology, we cannot trust the Gaussian approximation, and the full numerical Fourier transform has to be used. The results are best presented in the form of the standardized variable

$$\xi = \frac{x - \langle\, x \,\rangle}{\sigma} \quad . \qquad (109)$$

In fig.13 we plot $\sigma f_2(x)$ as a function of ξ for various dimensions. For comparison, we also give the shape of the standard normal distribution under the same plotting convention.

For comparison purposes, the confidence levels implied by $f_2(x)$ for a given dimensionality are relevant, as we have discussed. We therefore give, in table

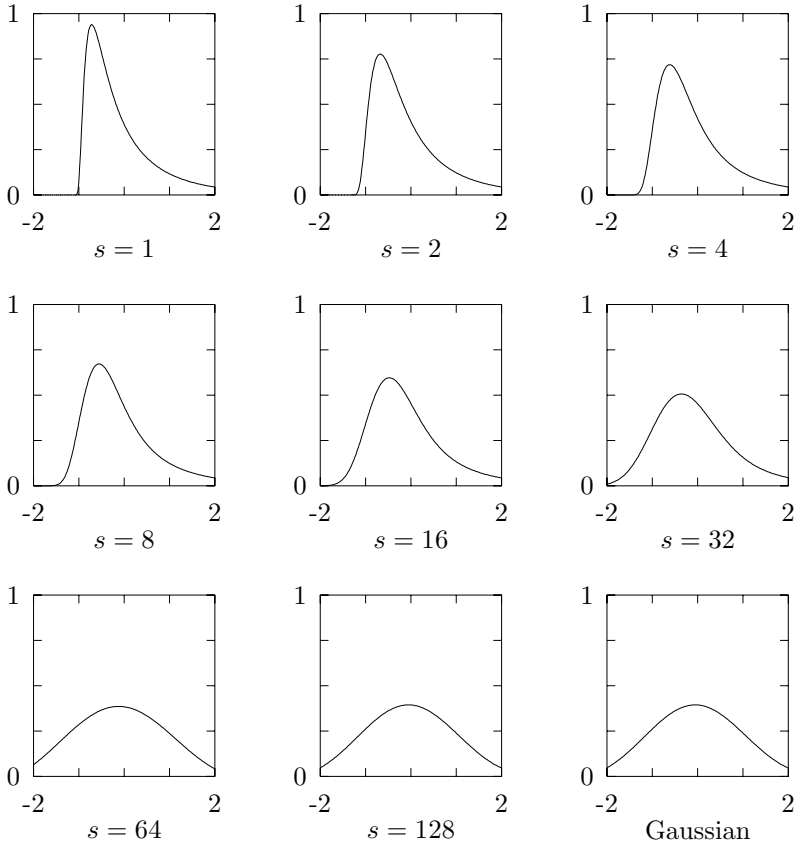


Figure 11: Probability density $\sigma f_2(\langle\, x \,\rangle + \sigma\xi)$ for the Wiener problem set, as a function of the standardized variable ξ, for dimensions from 1 to 128. The last plot is the normal distribution with the same convention.

s	mean	st.dev.	skewness
1	0.167E+00	0.149E+00	0.256E+01
2	0.139E+00	0.956E-01	0.239E+01
3	0.880E-01	0.502E-01	0.236E+01
4	0.502E-01	0.242E-01	0.234E+01
6	0.143E-01	0.491E-02	0.231E+01
8	0.375E-02	0.915E-03	0.226E+01
10	0.960E-03	0.163E-03	0.220E+01
12	0.242E-03	0.283E-04	0.214E+01
16	0.152E-04	0.819E-06	0.200E+01
20	0.953E-06	0.231E-07	0.186E+01
24	0.596E-07	0.647E-09	0.172E+01
32	0.233E-09	0.501E-12	0.147E+01
40	0.909E-12	0.387E-15	0.125E+01

Table 1: Parameters of $f_2(x)$ as a function of dimension.

2, a number of quantiles for a number of different dimensions. Note that the 50% quantile corresponds to the mode of the distribution, which (due to the skewness) is not necessarily equal to the mean value, which for the standardized variable ξ is of course zero. To have an impression of the numerical accuracy of the quantile determination, we also give the difference between 1 and the actual numerical value of the total integral, which ideally should be zero. The behaviour of the quantiles with dimension shows that the case $s = 1$ is not really typical, since the distribution for $s = 2$ appears to be much wider. This can be understood somewhat if we realize that $f_2(x)$ vanishes for negative x: if σ is comparable to $\langle\, x \,\rangle$, this forces the low quantiles to be relatively close to zero.

5.4 Orthonormal case

We now turn to the case of a problem set defined in terms of orthonormal functions, such as the Fourier or Walsh set. In these cases, there is no problem with factorization in higher dimensions. For $s = 1$, we have

$$\alpha(y_1, y_2) = \sum_{n \geq 1} \sigma_n^2 u_n(y_1) u_n(y_2) \quad ; \tag{110}$$

from which it is trivial to prove that

$$Z_m = \sum_{n \geq 1} \sigma_n^{2m} \quad . \tag{111}$$

Now, we may use $C_n = Z_n$, and $O_n = 0$, to construct the moment-generating function:

$$G_2(z) \quad = \quad \exp(\psi(z)) \quad ,$$

quantile	dimensionality							
	1	2	4	8	16	32	64	∞
0.1 %	-1.00	-1.15	-1.27	-1.39	-1.66	-2.06	-2.53	-3.09
1 %	-0.95	-1.06	-1.14	-1.23	-1.40	-1.67	-1.97	-2.33
5 %	-0.87	-0.94	-0.98	-1.03	-1.12	-1.28	-1.46	-1.64
10 %	-0.81	-0.86	-0.88	-0.90	-0.96	-1.06	-1.17	-1.28
50 %	-0.32	-0.29	-0.28	-0.26	-0.22	-0.16	-0.09	0.00
90 %	1.21	1.22	1.21	1.21	1.21	1.22	1.25	1.28
95 %	1.98	1.96	1.95	1.94	1.90	1.83	1.74	1.64
99 %	3.87	3.80	3.78	3.74	3.63	3.36	2.93	2.33
99.9 %	6.72	6.56	6.54	6.43	6.25	5.70	4.75	3.09
	7.6E-5	6.3E-5	6.0E-5	5.1E-5	4.1E-5	1.7E-5	2E-6	0

Table 2: Quantiles of the standardized distribution of the quadratic discrepancy for the Wiener problem set, for various dimensions. The infinite dimensionality corresponds to the normal distribution. The last line gives the accuracy of the total integral.

$$\begin{aligned}\psi(z) &= -\frac{1}{2}\sum_{n\geq 1}\log\left(1-2z\sigma_n^2\right) \\ &= -\frac{1}{2}\sum_{n\geq 1}^{K}\left[\log\left(1-2z\sigma_n^2\right)+2z\sigma_n^2+2z^2\sigma_n^4\right]+z\sum_{m\geq 1}\sigma_n^2+z^2\sum_{m\geq 1}\sigma_n^4 \quad ,\end{aligned} \tag{112}$$

where we have also indicated the proper way to evaluate it. In the more-dimensional case, we might assume that

$$\sigma_{\vec{n}} = \sigma_{(n_1,n_2,\ldots,n_s)} = \sigma_{n_1}\sigma_{n_2}\cdots\sigma_{n_s} \quad , \tag{113}$$

and obtain an analogous formula. An attractive choice would be to have

$$\sigma_n = \frac{1}{2n-1} \quad , \tag{114}$$

so that we may use the factor multiplicity $P_s(n)$ again:

$$\begin{aligned}G_2(z) &= \exp(\psi(z)) \quad , \\ \psi(z) &= -\frac{1}{2}\sum_{n\geq 1}P_s(n)\log\left(1-2z\sigma_n^2\right) \\ &= -\frac{1}{2}\sum_{n\geq 1}^{K}P_s(n)\left[\log\left(1-2z\sigma_n^2\right)+2z\sigma_n^2+2z^2\sigma_n^4\right]+z\xi(2)^s+z^2\xi(4)^s \quad ,\end{aligned} \tag{115}$$

6 Discrepancy Calculations

6.1 Extreme Discrepancy

Numerical calculation of the extreme discrepancy of a particular point set requires first calculating the local discrepancy at a position in the hypercube, then finding the maximum of that local discrepancy over the hypercube. The local discrepancy is calculated in time Ns, where N is the number of points and s the dimensionality, since it involves essentially counting the number of points all of whose coordinates are less than the position being considered. For high dimensionality some time can be saved because as soon as one coordinate of a point is greater than the coordinate of the position, it is not necessary to test the others, so the actual time would behave more like $N\sqrt{s}$. The major problem is of course finding the maximum of this function. The local discrepancy is discontinuous not only at every point, but also at every position of which any coordinate is the coordinate of a point. This means there are N^s candidate positions where the maximum could potentially occur. Again a clever program can save some time because some of these positions are not in fact possible maxima, but still the overall computational complexity can be expected to behave approximately as $N\sqrt{s}N^s$, which makes it prohibitive to calculate for large point sets in high dimensions. Practical limits are around ten points in ten dimensions or 100 points in five dimensions, which are too small to be of interest for real calculations.

6.2 Exact Quadratic Discrepancy

Although it may appear that quadratic discrepancy is more complicated than extreme discrepancy because it is defined as an integral over the hypercube, in fact the integrals involved are only step functions and it is possible to perform considerable simplification to reduce the calculation time. Perhaps the best way to explain how the calculation works is simply to give the Fortran code:

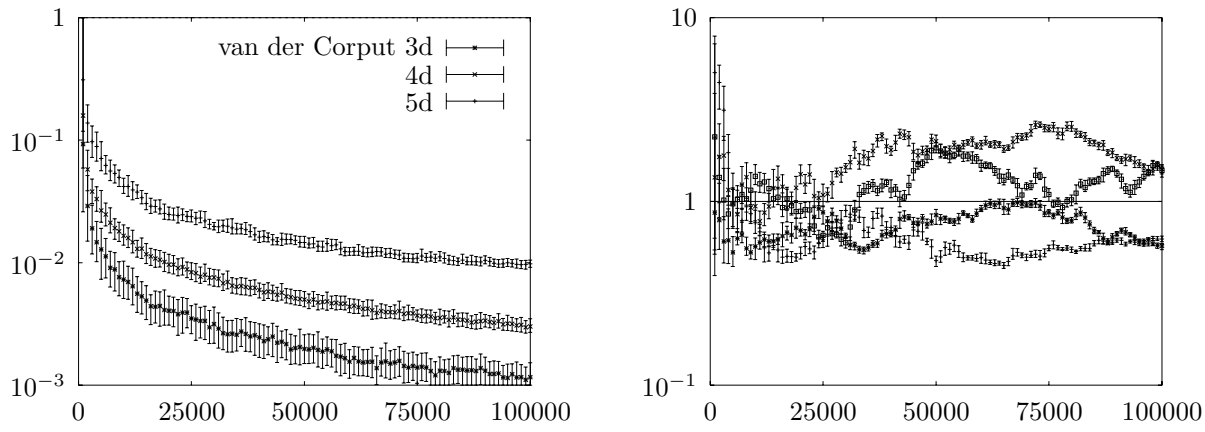
```
      subroutine discd2(ntot,ns,x,d2)
************************************************************
* compute the quadratic discrepancy for the point set
* x(i,mu), i=1,2,...,ntot    mu=1,2,...,ns
* the output is the array d2(n), n=1,2,...,ntot
* and gives the discrepancy * n^2 for the first n points
************************************************************
      implicit real*8(a-h,o-z)
      dimension x(ntot,ns),d2(ntot)
* initialize a few constants
      c2 = (1d0/2d0)**ns
      c3 = (1d0/3d0)**ns
      bn = 0
* start loop over number of points
      do 999 n = 1,ntot
        if(mod(n,100).eq.0) print *,'doing',n,'...'
```

```
* compute b(n) and a(n,n) for the new point
        a = 1.d0
        b = 1.d0
        do 1 mu = 1,ns
          a = a*(1.d0-x(n,mu))
          b = b*(1.d0-x(n,mu)**2)
    1   continue
        b = c2*b
* update running value of sum_b
        bn = bn+b
* case n=1
        if(n.eq.1) then
          d2(n) = a - 2*bn + c3
* case n>1
        else
* sum the a(i,n) for i=1 to n-1
          an = 0.d0
          do 3 i = 1,n-1
            temp = 1.d0
            do 2 mu = 1,ns
              temp = temp*(1.d0-dmax1(x(i,mu),x(n,mu)))
    2       continue
            an = an+temp
    3     continue
* give d2(n) for n>1 by relating it to d2(n-1)
          d2(n) = d2(n-1) + 2*an + a - 2*bn -2*(n-1)*b + (2*n-1)*c3
        endif
* end of loop over n
  999 continue
      end
```

Note that we used double precision to calculate the discrepancy, even though we generated the points in single precision. This is necessary since considerable precision is lost in the big sums, and we found that in single precision only about one or two digits of accuracy remained for the larger point sets.

6.3 Quadratic discrepancy by Monte Carlo

Since the quadratic discrepancy is a multidimensional integral, it can be estimated using a Monte Carlo approximation. If the desired accuracy is, for example 5% , we found Monte Carlo to be faster than the exact calculation for more than about 50,000 points. We used ordinary pseudorandom numbers for these calculations for fear that correlations between the two quasi-random point sets would produce unreliable results. We repeated some Monte Carlo calculations using the exact algorithm to verify that the agreement was good.



(a) van der Corput, $s = 3, 4$ and 5

(b) RanLux, $s = 4$

Figure 12: The quadratic discrepancy for up to 100 000 points for (a) the van der Corput generator in 3, 4 and 5 dimensions; and (b) four samples of random points in 4 dimensions.

6.4 Numerical Results

We have calculated the quadratic discrepancy for four different quasi-random generators: Richtmyer, van der Corput-Halton, Sobol' and Niederreiter-base-two, all from one to twenty dimensions. Exact calculations were made for 1-100,000 points, and Monte Carlo calculations for 100,000 and 150,000 point sets.

The programs used to generate the Richtmyer points and van der Corput points were written by us, and work in any number of dimensions limited only by some dimension statements and the table of prime numbers. The Sobol' and Niederreiter programs were kindly provided by Paul Bratley and Bennet Fox. Their Sobol' program is implemented for up to 40 dimensions, and we have extended their Niederreiter-base-two programs to work up to 25 dimensions.

We have consumed weeks of computer time producing dozens of plots of quadratic discrepancy as a function of N for the four quasi-random generators and also some good pseudorandom generators, in all dimensions from one to 20. It would be fastidious to show all these plots here, especially since many of them are rather similar and most of the behaviour can be summarized in some more revealing plots as we later discovered. However, we do show here a few typical plots as well as those which we found the most surprising.

Figure 12 shows some examples of quadratic discrepancy calculated for random and quasi-random point-sets up to 100 000 points. In these plots, the error bars give the total range of quadratic discrepancies calculated for all the 1000

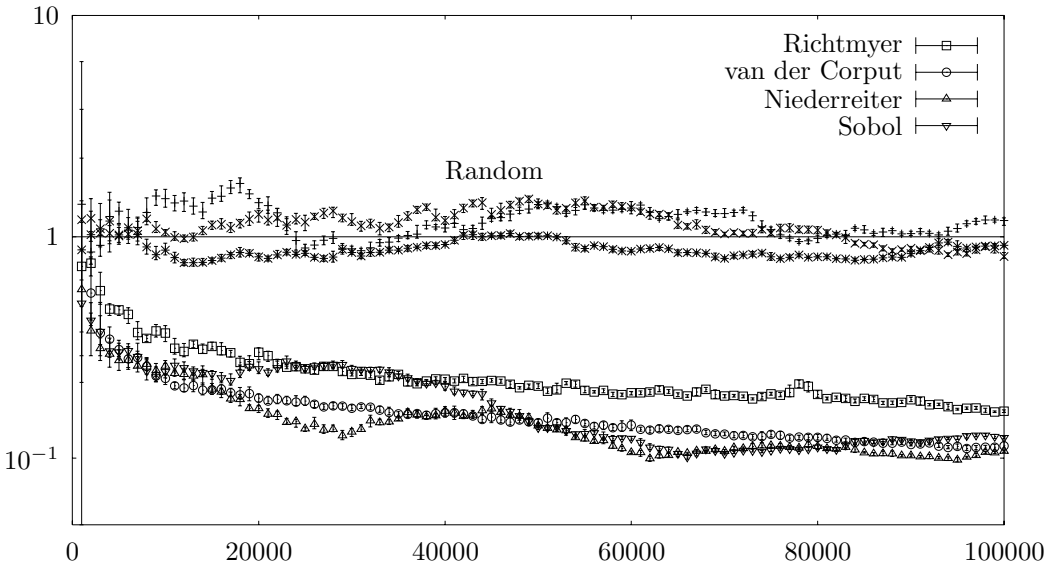


Figure 13: The quadratic discrepancy for up to 100 000 points for both random and quasi-random points in 8 dimensions. Three different random point samples are compared with our four quasi-random generators which all have much smaller discrepancy.

values of N in the range around the plotted value. Thus the tops and bottoms of the error bars give the envelope of the full plot, had we plotted a point for each value of N. Subfigure (a) shows a typical quasi-random behaviour for low dimensionality, in this case the van der Corput generator for 3, 4 and 5 dimensions. One sees that apart from very small point-sets, there is a considerable improvement over the expectation for random points (defined here as =1.0), and this improvement increases (faster convergence) with increasing N, as expected from the theoretical results. However, with increasing dimension, the improvement over random points decreases. For comparison, figure 12(b) shows the same quadratic discrepancy for four typical random point sets. Note the different scale, however, as these discrepancies are much larger than those of quasi-random points.

Figure 13 shows the discrepancy in eight dimensions for both random and quasi-random points. Our normalized values of quadratic discrepancy for three random sequences are of course close to one as they must be. All the four quasi-random sequences show a much lower discrepancy, between a factor five and ten lower for 50 to 100 000 points. For more than 40 000 points, the Richtmyer generator is not quite as good as the other three quasi-random generators. Note that these results are obtained for dimension 8. The results for other low dimensionalities are qualitatively similar.

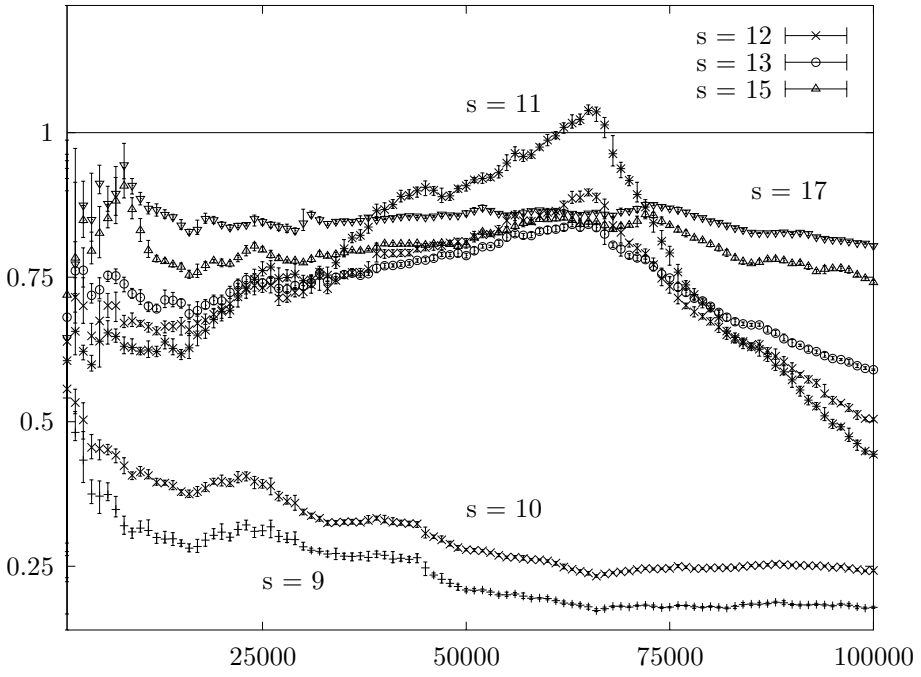
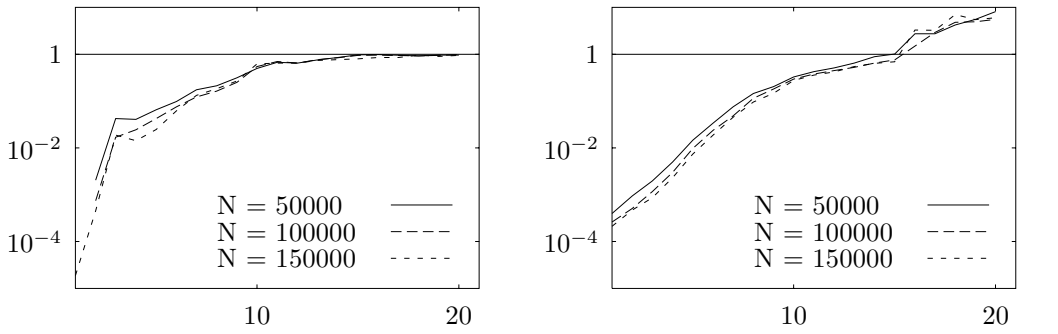


Figure 14: The quadratic discrepancy for up to 100 000 points for the Sobol' sequence in dimensions 9 to 17.

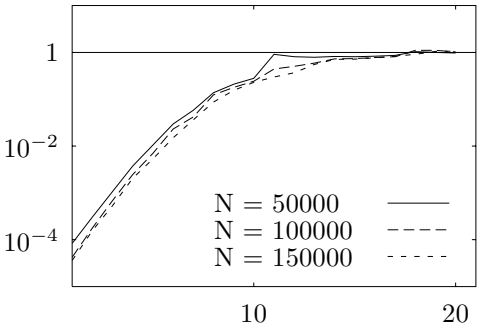
It must be realized, however, that for higher dimensionalities surprises may be in store. As an illustration, we give in figure 14 the discrepancy for the Sobol' sequence in dimensions 9 to 17, as a function of the number of points. It is seen that around 70,000 points the discrepancy it not small at all, and is actually worse than random for $s = 11$; after this bump, however, the discrepancy quickly returns to a very small value. Although such a behaviour can presumably be well understood from the underlying algorithm, it must be taken as a warning that for quasi-random sequences, an increase in the number of points does *not* automatically lead to an improvement. Note that this behaviour is not necessarily the same as the well-known 'irregularity of distribution', since in figure 14 it is actually the envelope of the discrepancy curve that shows a global, long-term rise (for $s = 11$, running from $N \sim 30,000$ to $N \sim 70,000$, a much larger scale than ought to be expected for *local* increases in discrepancy).

Figure 15 shows a summary of the quadratic discrepancies for all our quasi-random generators. As usual, all discrepancy values are normalized to the expectation for a truly random point set. Since we found the behaviour with respect to N rather uninteresting for most cases, we have instead plotted here the discrepancy for a few fixed values of N as a function of the dimensionality

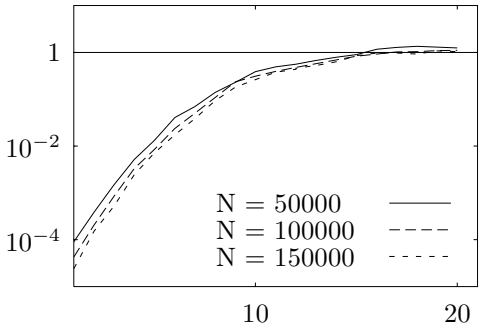


(a) Richtmeyer

(b) Van der Corput



(c) Sobol'



(d) Niederreiter

Figure 15: The quadratic discrepancy, normalized to the quadratic discrepancy for truly random numbers for N=50 000, 100 000 and 150 000 points as a function of dimensionality for four different quasi-random generators.

s. This is considerably more revealing, and in fact the first striking aspect is the resemblance between the behaviours of all the different generators.

Now we see that the dominant effect is that all generators are good for small s and all discrepancies rise sharply as a function of s, finally approaching the random expectation around $s = 15$ for $N = 150\ 000$.

The van der Corput generator becomes much worse than random above $s = 15$, whereas the other discrepancies remain rather close to random up to $s = 20$. Surprisingly, the only one that stays better than random up to $s = 20$ is the Richtmyer.

In Figure 16 we plot the discrepancies in terms of the variable ξ of Eq.(109). This representation ought to inform us about how 'special' the quasi-random sequences are compared to truly random ones, since on this scale 95% of all truly random sequences will fall between -2 and +2. Now we see that the lower discrepancy of the Richtmyer sequence compared to random sequences is really significant up to 20 dimensions, provided one uses more than 100 000 points. The poor behaviour of the Van der Corput sequences above $s = 15$ is highly significant, but the other generators look much like random ones between $s = 18$ and $s = 20$. This provides additional motivation to study the Richtmyer generator, which is easily implementable, and in particular to look for optimal constants S_μ.

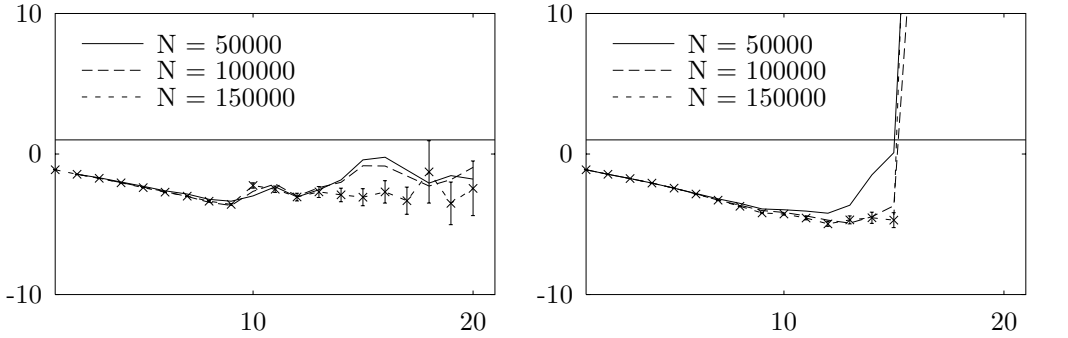
It should be noted, that the discrepancy improves in general with increasing numbers of points. It might be conjectured, that asymptotic behaviour of sequences' discrepancy will, for $s > 10$, only become evident for a larger number of points.

7 Conclusions

We have reviewed a number of arguments that suggest that the use of quasi-random numbers in multi-dimensional integration may lead to an improved error over classical Monte Carlo. The central notion in this context is that of discrepancy. We have shown, that there exist a relation between the definition of a discrepancy and the class of integrands of which our particular integrand is supposed to be a typical member. We have discussed several such classes and derived the appropriate induced discrepancies.

Another important aspect of discrepancy is the fact, that for a quasi-random point set it ought to be smaller than for a truly random point set. We have therefore studied the distribution of the value of discrepancy for truly random points.

Finally, we have computed the quadratic Wiener discrepancy for a number of quasi-random point sets. For all the quasi-random generators tested, the discrepancy, normalized to that of truly random numbers increases with increasing dimensionality and approaches that of truly random point sets around $s = 15$. If the quadratic discrepancy is plotted in terms of the standardized variable ξ, it is seen that the quadratic discrepancy for the quasi-random point sets is indeed significantly better than that of a typical member of the ensemble of



(a) Richtmeyer

(b) Van der Corput

(c) Sobol

(d) Niederreiter

Figure 16: The standardized quadratic discrepancy ξ for N=50 000, 100 000 a nd 150 000 points as a function of dimensionality for four different quasi-random generators. For $N = 150\ 000$, the Monte Carlo error-estimate is given.

random point sets for dimensions up to $s \approx 12$ for $N = 100\ 000$ to $s \approx 15$ for $N = 150\ 000$. Only one quasi-random generator (the Richtmyer) remains significantly better than random up to the highest dimension we calculated (twenty). For higher dimensions, it is conjectured that the asymptotic regime for the discrepancy is approached only for a much higher number of points N, which we did not cover in our computations.

Appendix A: Van der Corput discrepancy

Here we present some considerations on the behaviour of the van der Corput sequence in base 2, that can be obtained in a straightforward manner. We consider the extreme discrepancy for the finite N-member point set $x_k = \phi_2(k)$, with $k = 0, 1, 2, \ldots, N-1$, and we define

$$d(N) = N D_\infty(N) \quad . \tag{116}$$

We trivially have that $d(N) \geq 1$; more precisely, by inspection of the behaviour (given in fig.1) we observe the following recursion:

$$\begin{aligned} d(0) &= 0 \quad , \\ d(N) &= \begin{cases} d(\frac{N}{2}) & \text{if } N \text{ is even,} \\ \\ \frac{1}{2}\left(1 + d(\frac{N-1}{2}) + d(\frac{N+1}{2})\right) & \text{if } N \text{ is odd.} \end{cases} \end{aligned} \tag{117}$$

This recursion sheds some light on how the discrepancy depends on the binary properties of N, and in this manner we can compute $d(N)$ very quickly even for extremely large N. Now, we find the maximal discrepancy in the following way: in each interval between $N = 2^{p-1}$ and $N = 2^p$, we search for the maximum of $d(N)$. This is reached for two values of N, and we concentrate on the smallest one of these, which we denote by ν_p: the corresponding value $d(N)$ we denote by d_p. Again by inspection, we find the following recursions:

$$\nu_p = 2\nu_{p-1} + (-1)^p \quad , \quad d_p = \frac{1}{2}\left(d_{p-1} + d_{p-2} + 1\right) \quad , \tag{118}$$

with starting values $\nu_1 = 1$, $d_1 = 1$, and $d_2 = 3/2$. These relations are easily solved to give closed formulae for ν_p and d_p:

$$\nu_p = \frac{1}{3}\left(2^{N+1} + (-1)^p\right) \quad , \quad d_p = \frac{1}{9}\left(3N + 7 - (-2)^{1-N}\right) \quad . \tag{119}$$

Now, as N becomes large, we can easily eliminate p, and we find the asymptotic behaviour

$$d_p \sim \frac{\log \nu_p}{\log 8} \quad , \tag{120}$$

which agrees nicely with Eq.(26). For other bases, recursion relations similar to Eq.(117) ought to exist, but we have not pursued this.

Appendix B: completeness of the Fourier set

We present a short and simple proof of the completeness of the orthonormal functions $u_n(x)$ introduced in Eq.(58). To this end, consider the sum

$$\Delta_N(x, y) = \sum_{n=0}^{2N} u_n(x) u_n(y)$$

$$
\begin{aligned}
&= 1 + 2\sum_{n=1}^{N} [\cos 2\pi nx \cos 2\pi ny + \sin 2\pi nx \sin 2\pi ny] \\
&= 1 + 2\sum_{n=1}^{N} \cos 2\pi n(x-y) \\
&\equiv \Delta_N(x-y) \quad . \qquad (121)
\end{aligned}
$$

We consider the limit

$$
\Delta(\xi) = \lim_{N\to\infty} \Delta_N(\xi) \quad . \qquad (122)
$$

Obviously, this diverges for $\xi = 0$, and it remains to prove that we get zero if ξ is not zero. But we have, for such ξ,

$$
\begin{aligned}
\Delta(\xi) &= 1 + \sum_{n>0} \mathrm{Re} \exp(2i\pi\xi n) \\
&= 1 + 2\mathrm{Re} \sum_{n>0} (e^{2i\pi\xi})^n \\
&= 1 + 2\mathrm{Re} \frac{e^{2i\pi\xi}}{1 - e^{2i\pi\xi}} \quad , \qquad (123)
\end{aligned}
$$

which is indeed zero. Therefore, we have

$$
\Delta(\xi) = \delta(\xi) \quad , \qquad (124)
$$

the Dirac delta distribution. Note that we have to assume that the sum over n in Eq.(121) has to have an upper limit $2N$ rather than N: the way in which the limit is approached is important. Using the property that for a continuous function $f(x)$,

$$
\begin{aligned}
f(x) &= \int dy\ \delta(x-y)\ f(y) \\
&= \sum_{n\geq 0} u_n(x) \int dy\ u_n(y) f(y) \quad , \qquad (125)
\end{aligned}
$$

we immediately get the decomposition of $f(x)$ into the orthonormal base.

Appendix C: completeness of the Walsh set

We also prove that the set of all Walsh functions is complete. Let us first note that

$$
W_n(x) W_n(y) = W_n(\xi) \quad , \qquad (126)
$$

where ξ's binary expansion $0.\xi_1\xi_2\xi_3\cdots$ is the bitwise `XOR` of the binary expansions of x and y, that is, ξ has zeroes in all positions except those where the

binary digits of x and y are different. We now define

$$\begin{aligned}\Delta_N(x,y) &= \sum_{n=0}^{2^{N+1}-1} W_n(x)W_n(y) \\ &= \sum_{n=0}^{2^{N+1}-1} W_n(\xi) \\ &= \sum_{n_1=0}^{1}(-1)^{\xi_1 n_1}\sum_{n_2=0}^{1}(-1)^{\xi_2 n_2}\cdots\sum_{n_N=0}^{1}(-1)^{\xi_N n_N} \quad . \end{aligned} \tag{127}$$

Now, if n runs from 0 to $2^{N+1}-1$, all its binary digits $n_{1,2,\ldots,N}$ will be 0 and 1 an equal number of times. Therefore, each factor in the sum (127) will evaluate to zero unless its corresponding digit of ξ is also zero. Hence, $\Delta_N(x,y)$ is zero unless the binary digits of x and y agree up to and including the N^{th} place, and in that case it will be 2^N. Taking the limit $N\to\infty$ we again recover the Dirac delta distribution. As in the Fourier case, the way we take the limit turns out to be relevant.

Appendix D: The function $P_s(n)$

In this appendix we discuss the function $P_s(n)$, which counts in how many ways an odd integer n can be written as a product of s integers (including ones):

$$P_s(n) = \sum_{n_{1,2,\ldots,s}\geq 1} \delta_{n_1 n_2\cdots n_s\ ,\ n} \quad . \tag{128}$$

Obviously, $P_1(n)=1$, and we have the recursion

$$P_{s+1}(n) = \sum_{m\geq 1} P_s(m)\delta_{n\bmod m\ ,\ 0} \quad . \tag{129}$$

Note that the fact that n is odd automatically restricts these sums to the odd integers only. The recursion (129) can trivially be implemented to give us $P_s(n)$ for very large values of s and n. In the table we give its first few values. Note the irregular behaviour of $P_s(n)$ with n: for n prime it equals s, while new maxima are encountered whenever n is a product of the first consecutive odd numbers, or a triple of that number. Then, it can become quite large: for instance, $P_{10}(945)=22000$. We need, however, not worry about the convergence of series like those giving $\psi(z)$ and $\chi(z)$; for, we have

$$\sum_{n\geq 1} P_s(2n-1)\frac{1}{(2n-1)^x} = \left[\sum_{n\geq 1}\frac{1}{(2n-1)^x}\right]^s = \xi(x)^s \quad , \tag{130}$$

and sums like these are finite whenever $x>1$. The function $P_s(n)$ *without* the restriction to odd integers is discussed, for instance, by Hardy and Wright [2].

n	s									
	1	2	3	4	5	6	7	8	9	10
1	1	1	1	1	1	1	1	1	1	1
3	1	2	3	4	5	6	7	8	9	10
5	1	2	3	4	5	6	7	8	9	10
7	1	2	3	4	5	6	7	8	9	10
9	1	3	6	10	15	21	28	36	45	55
11	1	2	3	4	5	6	7	8	9	10
13	1	2	3	4	5	6	7	8	9	10
15	1	4	9	16	25	36	49	64	81	100
17	1	2	3	4	5	6	7	8	9	10
19	1	2	3	4	5	6	7	8	9	10
21	1	4	9	16	25	36	49	64	81	100
23	1	2	3	4	5	6	7	8	9	10
25	1	3	6	10	15	21	28	36	45	55
27	1	4	10	20	35	56	84	120	165	220
29	1	2	3	4	5	6	7	8	9	10
31	1	2	3	4	5	6	7	8	9	10
33	1	4	9	16	25	36	49	64	81	100
35	1	4	9	16	25	36	49	64	81	100
37	1	2	3	4	5	6	7	8	9	10
39	1	4	9	16	25	36	49	64	81	100
41	1	2	3	4	5	6	7	8	9	10
43	1	2	3	4	5	6	7	8	9	10
45	1	6	18	40	75	126	196	288	405	550

Table 3: Values of $P_s(n)$ for low values of s and n.

Acknowledgments

We are grateful to Paul Bratley and Bennet Fox for supplying several programs for quasi-random number generation and for advice on their usage.

References

[1] J.G. van der Corput, Nederl. Akad. Wetensch. Proc. ser. B 38 (1935) 813, 1058;
J.H. Halton, Numer. Math. 2(1960) 84; Err. 196.

[2] G.H. Hardy and E.M. Wright, An introduction to the theory of numbers, Oxford University Press, London, 1960

[3] F. James, Monte Carlo Theory and Practice, Rep. Prog. Phys. 43 (1980) 1145-1189.

[4] R. Kleiss, Average-case Complexity Distributions: A Generalization of the Woźniakowski Lemma for Multidimensional Numerical Integration, Comp. Phys Comm. 71(1992) 39-46.

[5] J. Hoogland,R. Kleiss, Discrepancy-based error-estimates for quasi-Monte Carlo. 1:General formalism To be published in Comput.Phys.Commun. e-Print Archive: http://xxx.lanl.gov/hep-ph/9601270
J. Hoogland,R. Kleiss, Discrepancy-based error-estimates for quasi-Monte Carlo. 2:Results for one dimension To be published in Comput.Phys.Commun. e-Print Archive: http://xxx.lanl.gov/hep-ph/9603211

[6] Kuipers and Niederreiter, Uniform Distribution of Sequences (Wiley, New York, 1974)

[7] H. Niederreiter, Random Number Generation and Quasi-Monte Carlo Methods (SIAM, Philadelphia, 1992)

[8] M. Lüscher, A portable high-quality random number generator for lattice field theory simulations, Comp. Phys. Comm. 79 (1994) 100;
F. James, RANLUX: A Fortran implementation of the high-quality pseudorandom number generator of Lüscher, Comp. Phys. Comm. 79 (1994) 111.

[9] M. Berblinger and Ch. Schlier, Monte Carlo integration with quasi-random numbers: some experience, Comp. Phys. Comm. 66(1991) 157-166

[10] I.M. Sobol', USSR Comput. Maths. Math. Phys. 7 (1967) 86;
I.M. Sobol', USSR Comput. Maths. Math. Phys. 16 (1977) 236-242;
P. Bratley and B.L. Fox, ACM Trans. Math. Software 14 (1988) 88.

[11] H. Woźniakowski, Bull. AMS 24 (1991) 185.

[12] S. Paskov, Journal of Complexity, Vol. 9, p. 291-312, 1993.